NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

TOWARD FINDING DRIVING COMMUNICATIONS FACTORS IN THE SYSTEM OF SYSTEMS SURVIVABILITY SIMULATION MODEL

by

Joseph R. Cordell

March 2014

Thesis Advisor: Dashi I. Singham
Second Reader: Thomas W. Lucas

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704–0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2014	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE TOWARD FINDING DRIVING COMMUNICATIONS FACTORS IN THE SYSTEM OF SYSTEMS SURVIVABILITY SIMULATION MODEL			**5. FUNDING NUMBERS**
6. AUTHOR(S) Joseph R. Cordell			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943–5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Army Research Laboratory, 2800 Powder Mill Road, Adelphi, Maryland			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ___N/A_____.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release;distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

The System of Systems Survivability Simulation (S4) was created by the Army Research Laboratory's Survivability and Lethality Analysis Directorate in cooperation with the New Mexico State University Physical Science Laboratory. S4 is a multi-level, agent-based, time-stepped, high resolution, stochastic combat model with a focus on survivability and lethality of equipment and forces. There are over 300 factors (or input parameters) used to define the elements on the simulated battlefield. This thesis explores a factor screening method using a supersaturated design that could be used to eliminate insignificant design parameters for given scenarios. Eliminating insignificant parameters could reduce the run-time of an experiment, thereby allowing a more robust design to be used only on the significant factors that are selected. The ability of the method to properly identify significant parameters is analyzed by creating a model in which the significant factors are already known and determining how well the method identifies the significant factors. The results of the analysis show that the method is effective when the factors are moderately to highly significant and for a small number of significant factors. Additional research comparing this method with other factor screening methods may lead to the use of this method when there are more factors than design points.

14. SUBJECT TERMS			**15. NUMBER OF PAGES**
System of Systems, S4 System of Systems Survivability Simulation, Design of Experiments, Nearly Orthogonal Latin Hypercube, Nearly Orthogonal Nearly Balanced Mixed Design, Factor Screening, Survivability Lethality and Vulnerability Assessment, Supersaturated Design, Stepwise Regression			77
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540–01–280–5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

TOWARD FINDING DRIVING COMMUNICATIONS FACTORS IN THE SYSTEM OF SYSTEMS SURVIVABILITY SIMULATION MODEL

Joseph R. Cordell
Captain, United States Army Reserve
B.S., Miles College, 2000

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 2014

Author: Joseph R. Cordell

Approved by: Dashi I. Singham
Thesis Advisor

Thomas W. Lucas
Second Reader

Robert Dell
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The System of Systems Survivability Simulation (S4) was created by the Army Research Laboratory's Survivability and Lethality Analysis Directorate in cooperation with the New Mexico State University Physical Science Laboratory. S4 is a multi-level, agent-based, time-stepped, high resolution, stochastic combat model with a focus on survivability and lethality of equipment and forces. There are over 300 factors (or input parameters) used to define the elements on the simulated battlefield. This thesis explores a factor screening method using a supersaturated design that could be used to eliminate insignificant design parameters for given scenarios. Eliminating insignificant parameters could reduce the run-time of an experiment, thereby allowing a more robust design to be used only on the significant factors that are selected. The ability of the method to properly identify significant parameters is analyzed by creating a model in which the significant factors are already known and determining how well the method identifies the significant factors. The results of the analysis show that the method is effective when the factors are moderately to highly significant and for a small number of significant factors. Additional research comparing this method with other factor screening methods may lead to the use of this method when there are more factors than design points.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ARL-SLAD	Army Research Laboratory Survivability and Lethality Analysis Directorate
B2P2	Brigade and Below Propagation and Protocol
BN	battalion
BN CDR	battalion commander
BSAB	blue situational swareness of blue
BSAR	blue situational awareness of red
C2	command and control
CDR	commander
CO	company
CO CDR	company commander
CSV	comma separated value
DMP	decision-making process
DMSC	Dynamic Model of Situated Cognition
DOE	design of experiments
LDR	leader
MOE	measure of effectiveness
MOP	measure of performance
NMSU-PSL	New Mexico State University Physical Science Laboratory
NOLH	nearly orthogonal Latin hypercube
NONBMD	nearly orthogonal nearly balanced mixed design
PLT	platoon
PLT LDR	platoon leader
S4	System of Systems Survivability Simulation
SA	situational awareness
SLV	survivability, lethality, and vulnerability
SLVA	survivability, lethality, and vulnerability analysis
SoS	system of systems
SoSA	system of systems analysis
SQD	squad

SQD LDR	squad leader
SSD	supersaturated design
VQ	voice quality
VV&A	verification, validation, and accreditation

EXECUTIVE SUMMARY

Traditionally, when acquisitioning a new system (e.g., armored vehicle, communications device, weapon system, etc.), the U.S. Army has compared the attributes of the system to be replaced and the new system. Research has shown that in order to conduct survivability, lethality, and vulnerability analysis (SLVA) on a new system, the new system should be tested as part of a larger system that includes all other equipment and platforms on the battlefield. Decision making attributes of those who operate the equipment and platforms and those that are in leadership roles should also be considered when conducting SLVA on a new system. This idea of analyzing a system as part of a larger system is known as system of systems analysis. Connecting all of the components of a system of systems (SoS) requires the components to be interconnected through a network creating a network-centric force. In order to increase survivability of a system, the communications environment for the network to which it belongs must be dependable. To analyze the survivability of a system within a SoS, a model must appropriately capture the physical attributes of the system and all of its components and allow the ability for an agent (i.e., soldier, platoon leader, company commander) to make or change decisions. The system of systems survivability simulation (S4) model was created by the Army Research Laboratory's Survivability and Lethality Analysis Directorate—along with the New Mexico State University Physical Science Laboratory—to model all aspects of the battlefield to include sensing, communications, maneuvering, engagement, ballistic damage, and agent decision making in order to conduct SLVA for a new system as part of an SoS.

The S4 model is composed of seven underlying models and contains approximately 300 input variables. With such a large number of parameters, it would take much effort and time to explore the model without using an efficient design. Designs that are based on the use of an orthogonal Latin hypercube allow an analyst to explore more of the parameter space while at the same time reducing the amount of time needed to conduct an experiment. Other design methods, such as the factor screening technique used in this thesis, allow potential influential parameters to be more quickly identified.

One benefit of using a factor screening technique is that the factor screening may allow a more focused design to be run on only the significant factors that are selected. This allows an analyst to further explore the model by reducing the number of parameters to be explored, which enables a smaller, less time-intensive DOE to be used.

This thesis sets out to test the ability of the factor screening method to identify key parameters. To do this, an experiment is created using a known function that acts as a generic model and produces a stochastic response. Within the model there are four components that are varied in the experiment that allow for the analysis of the factor screening method. The components used for the model are the number of significant factors, the mean of the random coefficients, the number of steps for the stepwise regression to use, and the standard deviation for the random noise. The response is then analyzed to see if the factor screening method is appropriately identifying the influential factors of the model.

The basic concept of the experiment is to randomly generate a vector of coefficients based on the number of significant factors and the mean of the random coefficients. The randomly generated vector of coefficients indicates the true significant factors. The vector of coefficients is then multiplied with the supersaturated design matrix to create a response vector. Random noise is then added to the response vector. Stepwise regression is then used to determine the significant factors based on the design matrix and the response vector. The significant factors chosen by the stepwise regression are then compared to the true significant factors.

For the analysis of the factor screening method, three responses are used. The responses include the probability of detecting all of the significant parameters, the proportion of significant parameters selected, and the probability of incorrectly assigning the wrong coefficient sign to a significant factor. The results of the analysis show that the probability of detecting all of the significant parameters varies based upon the mean of the coefficients for the significant factors and the number of significant factors. The number of steps used in the stepwise regression is not significant. The same results apply to the proportion of significant parameters selected. The factor screening method works well for factors that are moderately to highly significant, but not as much for those that

are only slightly significant. Additionally, as the number of significant factors increases, both the probability of detecting all of the significant parameters and the proportion of significant parameters selected decreases. The probability of incorrectly assigning the wrong coefficient sign to a significant factor mostly depends on the mean value of the coefficients. The probability of incorrectly assigning the wrong coefficient sign to a significant factor primarily occurs when the coefficients are only slightly significant and increases as the number of significant factors increases.

This research concludes that the factor screening method using a supersaturated design and stepwise regression may be beneficial in exploring models such as S4. Further research in this area will be needed to be able to apply the method to models other than the function that was used to test the method, as the supersaturated designs have to be created specifically for a given parameter space. It is recommended that additional research is conducted to compare results of the method with other factor screening methods which are known to be effective.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

This thesis has been a roller coaster ride, but thankfully my advisor, Dr. Dashi Singham, was able to motivate me to stay on track, though my target was not always known. Both Dr. Singham and my second reader, Dr. Tom Lucas, gave me great support during my times of need. Their advice and guidance were influential in enabling the completion of this thesis.

I would also like to thank the entire S4 team at NMSU-PSL and ARL-SLAD for their support. They allowed me to visit and have a firsthand look at the S4 model, answering all of my questions about the model in great detail. I would especially like to thank Dr. Jayashree Harikumar for keeping me informed and up to date with any information that I needed about the model and the issues that they were encountering.

Additionally, I would like to thank Bill Evans for helping me troubleshoot the R code that was written for the model. His expertise significantly reduced the amount of time needed to run the simulation.

I would like to give a very special thanks to Dr. Susan Sanchez, without whose support I may have never completed a thesis. Dr. Sanchez was able to provide, on short notice, an alternate thesis focus that did not require a complete overhaul of the work that I had already put into the writing of my original thesis.

Finally, I would like to thank my wife, Angie, for her support of me throughout my time at NPS. She has been my inspiration throughout my stay here. The sacrifices that she made were many and I would not be where I am today without her.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Traditionally, when acquisitioning a new type of equipment or platform (e.g., armored vehicle, communications device, weapon system), the U.S. Army has compared the attributes of the old and new equipment. In recent years, a change was made to not only look at the platforms and their capabilities as single units, but as a part of a larger system (Starks & Flores, 2004). One of the primary concerns for a new platform is its survivability, lethality, and vulnerability (SLV) as part of a larger force. To determine how effective a new platform is within a system of systems (SoS), the platform must be connected to a network of all the systems within the SoS (Bernstein, Flores, & Starks, 2006). This network includes integrating communications among all systems, and implementing a command and control (C2) structure that allows for better communications flow to help increase the survivability of each platform. In order to properly model the survivability of a platform within a SoS, the model must include this C2 hierarchy along with the ability for agents (i.e., soldier, platoon leader, company commander) to make or change decisions based on the information that they receive through the communications and organic sensing channels (Davidson & Pogel, 2010). In addition, the model must appropriately capture the physical attributes of the platform and all of its components in a way that is indicative of the actual environment to which the platform belongs.

The System of Systems Survivability Simulation (S4) is a model that has been created to focus on the SLV of a platform within a SoS (Davidson & Pogel, 2010). S4 models both the physical attributes of a platform and the decision-making processes (DMPs) of agents associated with the platform. S4 attempts to capture the critical aspects of the battlefield to include sensing, communications, maneuvering, engagement, ballistic damage, and agent decision making (Bernstein et al., 2006). Two of the key foci within S4 are the communications of entities within the network of systems and the DMPs that are associated with the entity (Davidson & Pogel, 2010).

S4 has been in development since 2004 and has yet to go through the verification, validation, and accreditation (VV&A) process. VV&A is the process that a model goes

through to make sure the results of the model are appropriate for the purpose for which the model was developed. The VV&A process serves as the quality control element of the simulation model. Without the accreditation, the model will not become an official tool used by the U.S. Army Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics (OUSD[AT&L]) (2009).

Each factor in the S4 model, whether continuous, discrete, or categorical, has a range of values or levels that may be varied to see the effects. This can be done by just looking at the extreme ranges of the factors, but looking at only the extreme ranges would only allow for the interpretation of linear effects in the model and would require 2^n design points, if all possible extreme combinations are explored, where n is the total number of factors. Another possibility is to include a central point for the range of each factor, but this increases the total number of combinations from 2^n to 3^n. If there are too many factors to consider simultaneously, it would be time consuming to run an experiment with all of the possible combinations of the extreme points of each factor, especially if a central point is included (Sanchez & Wan, 2009). Since each replication of the S4 model can take as long as 35 minutes, a smarter way of looking at the possible variations of the factors while reducing the amount of run time needed to produce results for analysis is desirable. This goal can be obtained by using design of experiments (DOE) and factor screening. Using DOE and factor screening will allow the S4 model to be investigated more effeciently. The factor screening technique may allow the analyst the ability to create a more focused design using only the significant factors that are selected by the technique. The use of DOE and factor screening, along with data analysis, may bring S4 closer to being ready for the VV&A process. These techniques would also be valuable in conducting S4's VV&A.

A. BACKGROUND AND LITERATURE REVIEW

In recent years, it has become evident that it is not likely that we will be fighting a conventional force, and that warfare will become more asymmetric as the U.S. Army is used more in operations other than war (Davidson, Pogel, & Smith, 2008, p. 154). Since 2001, the U.S. has been involved in asymmetric warfare as we have combated terrorism

around the globe. This asymmetric warfare, along with many technological advances, has led the Army into consideration of the 21st Century Strategic Environment to produce the Army's 2004 Transformation Roadmap (Davidson et al., 2008). The Army Transformation Roadmap describes the development of more rapidly deployable, more lethal, better informed, better protected, modular Brigade Combat Teams that are able to adapt to any environment in which they deploy (Davidson et al., 2008). Part of the transformation is to identify, articulate, and actively pursue survivability, lethality, and vulnerability analysis (SLVA) methods that can gauge how well forces are adapting to new environments (Starks & Flores, 2004). It has also become evident that in today's age information superiority is a force multiplier and allows a unit to gain a combat power advantage. In order for U.S. Forces to adapt more quickly, they depend on this information superiority to allow transmission of accurate and timely information to all decision makers on the battlefield (Starks & Flores, 2004). In addition to superiority of the information environment, leaders must be able to utilize the information that they receive to make proper tactical decisions based on their situational awareness. In order to properly model this, the model must include both the human and technological components of complex systems that illustrate how decision-making is influenced by information gathering and distribution (Miller & Shattuck, 2004).

The Dynamic Model of Situated Cognition (DMSC), introduced by Miller and Shattuck in 2003, models the way that information is gathered by sensors or technological means and how it is ultimately perceived by a decision-maker. DMSC offers a conceptual model of the technological and cognitive processes that lay the groundwork for how S4 models decision-making processes based on the information received through the organic sensing and communications channels (Hudak, Mullen, & Pogel, 2008). According to Starks and Flores (2004), there are three conditions necessary in order to properly conduct SLVA:

- Events and responses in a scenario cannot be scripted and must allow a decision-maker to dynamically change tactics or strategy based on their current situational awareness (SA)
- The model must have a network-centric or SoS approach that models organic sensing and communications

- Input parameters, output performance metrics, and functional relationships must be appropriate for SLVA in a SoS.

The agency responsible for developing an analysis tool for SLV is the Army Research Laboratory Survivability and Lethality Analysis Directorate (ARL-SLAD) based at White Sands Missile Range, New Mexico. One of the primary responsibilities of ARL-SLAD is to provide SLV assessments and information needed for senior leaders to make proper decisions about current and future force structure. With the assistance of the New Mexico State University Physical Science Laboratory (NMSU-PSL), ARL-SLAD created the S4 model, based on the DMSC and the three previous conditions, which explores not only the capabilities of a system, but also the communications network in which the platform belongs and the decisions made by agents using available information.

B. RESEARCH QUESTIONS

The intent of this thesis is to conduct analysis of the S4 model and provide NMSU-PSL and ARL-SLAD information on design of experiment and factor screening procedures that would significantly increase their productivity. The thesis is guided by the following questions:

1. What are the driving or most influential communications factors in the S4 model?
2. Given a supersaturated design (SSD) with a limited number of design points, can influential factors be properly identified using a stepwise regression factor screening technique?
3. How effective will a factor screening technique using stepwise regression be in identifying influential factors within the S4 model?

C. BENEFITS OF THE THESIS

Originally, this thesis was primarily focused on the analysis of output from the S4 model to determine the driving communications factors within the model, with a secondary goal of providing experimental design tools for future use by NMSU-PSL and ARL-SLAD. After the S4 team in New Mexico had multiple coding issues with S4, it was determined that there would not be sufficient time to complete the simulation and have enough time left for analysis. The focus was then changed to providing DOEs and a

stepwise regression factor screening technique that may be used to help identify important factors within the S4 model. Since the factor screening technique has never been used for the S4 model, it is used with a separate generic model to see if it is viable and thus potentially useful with S4. This research allows NMSU-PSL and ARL-SLAD more flexibility in creating their own designs. This flexibility will enable the S4 team to more effectively explore the S4 model. Additionally, this research provides NMSU-PSL and ARL-SLAD the opportunity to reduce the total runtime needed for their simulations by eliminating potentially unimportant factors prior to creating a study design.

D. METHODOLOGY

This thesis introduces a factor screening method that can be used for quickly identifying significant factors when there are many factors to consider, without expending great amounts of effort running time-intensive simulations. This is done by creating a DOE for a model in which significant factors are known, and then performing regression analysis to determine the ability of the factor screening method to identify important factors for different combinations of the model compnents. The created model has four components that are analyzed to determine the effectiveness and limitations of the factor screening technique.

THIS PAGE INTENTIONALLY LEFT BLANK

II. S4 MODEL OVERVIEW

The purpose or intent of the S4 model is to allow analysts to see how a platform or entity performs not just as a function of its capabilities, but with the addition of other platforms and entities as a part of a SoS. The analysts are looking for interactions over time and space to see how different systems perform together as part of a SoS. This will enable analysts to look at the SLV of a system within the context of a broad family of systems. The SLV analysis of a system can be used to identify issues with the system and allow the analysts to inform product managers of problems that will need to be corrected. Before any of this can be achieved, the model must be able to capture the critical aspects of the battlefield, including physical attributes of equipment, the battlefield environment, and dynamic decision-making of agents.

A. S4 DESIGN

The S4 model is a multi-level, agent-based, time-stepped, high resolution, stochastic combat model with a focus on survivability and lethality of equipment. S4 models forces ranging from teams, approximately five soldiers, to battalions, approximately 450 soldiers. The S4 model is composed of seven underlying models and contains approximately 300 input variables. S4 models physical attributes of personnel and equipment as well as the terrain in six of the seven underlying models. The remaining underlying model is used to represent the dynamic decision making of an agent. The underlying models are multi-level models that allow an action to take place. The levels of the underlying models are hierarchical and depend on the previous level of the underlying model. The first level of each of the underlying models is the device loops within the model. The second level shows individual devices that belong to the model or the instance execution of the individual devices. The last level of the underlying models is used to show the detailed execution of devices within the models (Hartley, 2013). Actions consist of movement, sensing, communication, ballistic engagement, damage after engagement, platform update, and agent decision-making (see Table 1).

Table 1. List of underlying models within the S4 environment and their respective function.

SUB-MODEL	BASIC MODEL FUNCTION
Sensor model	Models all sensors on all platforms and detection attempts for the sensors
Communications model	Models all communications devices on all platforms, including radio communications, data communications, and direct communication using voice or hand signals; Models communications jamming devices
Engagement model	Models engagement devices for all platforms and their performance to include the number of shots fired, direction and trajectory of each round, and ammunition status
Mobility model	Models mobility capabilities of all platforms
Damage model	Models damage to all platforms from ballistic engagement; damage can occur to any system that is part of the platform
Platform Status model	Updates the capabilities of all platforms (i.e., platform destroyed, weapon malfunction, driver injured, platform immobilized, etc.)
Agent model	Models the collective decision making capability of all soldiers associated with a platform, or of an individual dismounted soldier

At each time-step in the simulation, the underlying models are executed to allow for a possible change of SA for any entity within the simulation. Each of the underlying models provides information specifically for the aspect of the battlefield represented by the model. The cycle of execution within each frame or time-step is shown in Figure 1. After each sub-model is executed, the results of the sub-model are sent to the agent model as feedback. Based on the capabilities of the agent, information is passed to the next model in the chain to repeat the cycle until the S4 model reaches the execution of the agent model. At this time, the agent can make a decision based on the input from all of the other underlying models and perform the action that it decides based on its capabilities (Hartley, 2013). All of the modeled actions remain important in the model, but the two key components of the S4 model are the communications model and the agent model.

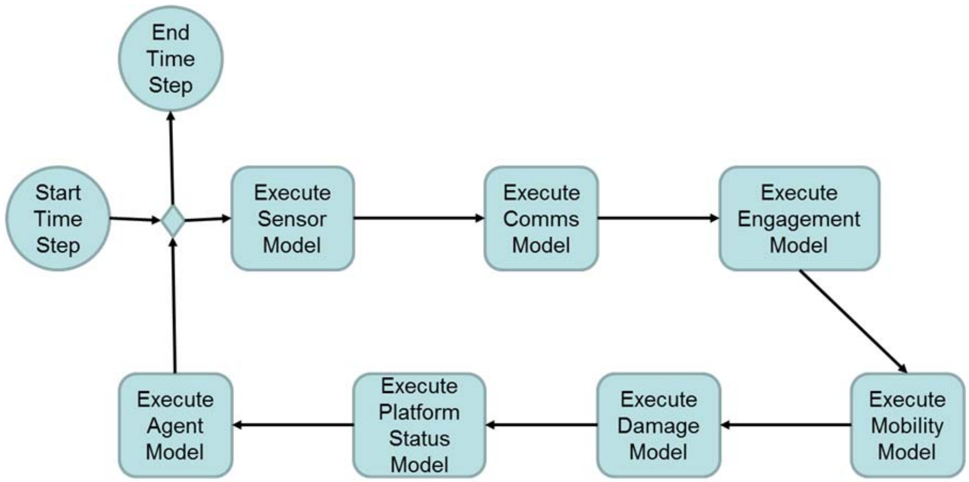


Figure 1. S4 model top-level flow chart.

B. COMMUNICATIONS MODEL

The communication model within S4 is a three-level model that handles all communications devices on all platforms. The first level, or top level, of the model allows agents to receive and process information within the same time-frame. The information in the top level is first received from the sensor model. The second level of the model is more detailed than the first. This is where platforms are provided with communications arrays that may consist of several radio devices. In this portion of the model, a platform receives the information from the sensor model for processing. The information from the sensor model includes information that the platform or agent receives by their own organic sensing capabilities, or through intelligence that is passed through the communications network as either voice or data information. This is how the communications arrays are built (Hartley, 2013). Once the information is received by the platform or agent, the communications device is then adjudicated to see if the device is working properly, and then the communications DMP is enabled. Once a decision has been made by the agent, the agent commences to deliver communication messages to other agents within the same communications network (see Figure 2).

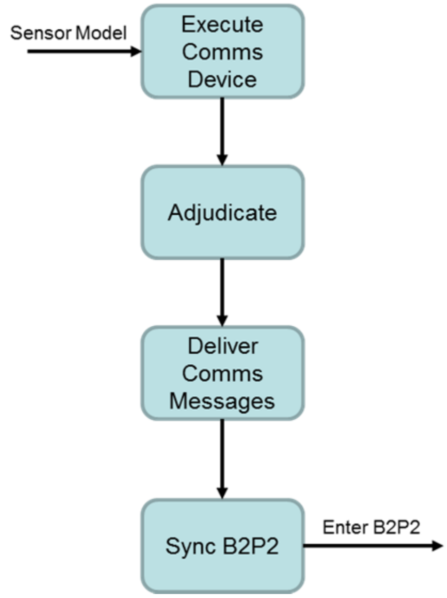


Figure 2. Level 2 communications model flow chart (from Hartley, 2013).

Once the agent makes a decision to send a message, the model enters the third level of the communications model. Processing of information or events occurs in a separate model, Brigade and Below Propagation and Protocol (B2P2). The B2P2 model is an event-based communication model, which uses Simkit for processing events (Hartley, 2013). Simkit is used for creating discrete event simulation (DES) models and was written by Professor Arnold Buss of the Naval Postgraduate School (Buss, 2002). The use of B2P2 requires synchronization with the S4 model as the S4 model is time-stepped. In order to accomplish this, the S4 model places a synchronization event 0.5 seconds into the future on the B2P2 event queue, which ensures that B2P2 does not get ahead of the S4 model. Synchronization events are used as placeholders to make the communications model wait for the next time step before processing events that occur in between the steps (Hartley, 2013).

C. AGENTS AND DECISION MAKING PROCESSES

Agents in the model are the decision-makers. They range from a dismounted soldier on the battlefield to the battalion commander. Within S4, the key leaders are the battalion commander (BN CDR), company commanders (CO CDRs), platoon leaders (PLT LDRs), squad leaders (SQD LDRs), and team leaders (TM LDRs). The PLs are the

most versatile and dynamic decision-makers in the S4 model since most action takes place at the platoon (PLT) level. The DMPs of the PLT LDR are the most complicated in the S4 model (Davidson & Pogel, 2010). PLT LDRs make tactical decisions based on the information they have using projection algorithms and task parameterizations which give execution details of the tasks. PLT LDRs take into account self-status, peers' status, terrain, and knowledge of the enemy when making projections. Using this information, the PLT LDRs create alternative scenarios for completing the mission and choose the best scenario to improve survivability and mission effectiveness (Bernstein et al., 2006). The DMPs for CO CDRs and BN CDRs use a library of situation response templates where the CDR makes decisions based on the current SA of the battlefield by scoring the information received against the library of templates. These scores are used with a consensus function that determines a best fit template for the situation (Davidson et al., 2008).

Agents are modeled as part of a hierarchy with all agents belonging to a specific command and control structure. Each agent also belongs to a communications network in which they can only intercommunicate with those who are part of their network. Agents have three basic components or characteristics. Agents have a specific set of capabilities, are able to move, and make decisions. When agents are part of a specific platform, such as an armored vehicle, the agents are modeled based on the agents' capabilities as well as the capabilities of the platform for which they belong (Davidson et al., 2008).

There are two types of agents modeled in S4. Objective agents are the actual agents that are modeled and they make decisions based on their current SA of the battlefield. Objective agents may not have perfect SA if information transmitted or sensed is lost or only partially received. The second type of agent is the subjective agent. Subjective agents are modeled with perfect SA based on information that the objective agent that they represent should have received. The decisions made by the two agents are compared, thereby allowing analysts to determine the impact of information that is imperfectly received by the objective agent through sensing or communications.

The DMPs in the S4 model for an agent are determined by the role that the agent plays within the simulation. DMPs for a SQD LDR are different than those of a CO CDR

and so on. As stated before, PLT LDRs have the most complicated and dynamic DMPs within S4. The DMPs for an agent are highly dependent on their current SA. Each DMP has a scripted set of rules. These rules include items such as the overall objective, target prioritization, and contingency planning. The DMPs are the core of the “sense, decide, and action loop” (Bernstein et al., 2006, p. 7). Agents cannot take any action until they have made a decision. Additionally, an agent cannot make a decision unless the proper information is received. Information can be received by an agent through its organic sensing capability or from other agents and platforms within its communication network; therefore, the DMPs for an agent are highly dependent on the success or failure of the communication network (Bernstein et al., 2006). To begin the simulation, agents must communicate their current status before they can proceed. Agents are given a mission or objective at the onset of the simulation through their C2 hierarchy. Over the course of the simulation, the agents may change their tactics or overall objective based on their current perception of the battlefield environment. “The degree to which an agent simply carries out orders or responds to the present situation can vary spatially over the battlefield and over time” (Bernstein et al., 2006, p. 6).

D. SYSTEM OF SYSTEMS ANALYSIS PROCESS

The first step in the traditional system of systems analysis (SoSA) process is the problem formulation phase. In this phase, the threat assessment, friendly force mix, and engagement environment must be specified as well as environmental factors. Tactics for the blue and red forces are also developed in this phase (Smith et al., 2012).

The next phase is the problem focus phase. In this phase, areas of uncertainty are identified and are added to the list of factors that will be used as inputs to the study. Additionally, a study question or questions will be identified. Once the problem has been conceptualized, the analysis plan is created using the input parameters, identifying performance measures that are of interest, usually measures of effectiveness (MOEs) or measures of performance (MOPs), and developing the scenario for the simulation (Smith et al., 2012).

After the parameters and their ranges have been identified and metrics have been specified, the input data is configured to run the simulation. This is where the traditional SoSA model should change to include DOE options. Prior to running any simulation, there should be time to study the benefits of alternative DOEs and to develop a DOE that would work best for the current configuration of the parameter space. When a proper design has been created for the parameter space, then the simulation will be conducted for the scenario (Smith et al., 2012).

Upon completion of the simulation, the data is then collected and analyzed using analytical and playback tools to answer the study questions using the MOEs or MOPs that were measured. Playback tools are used to look at individual simulation runs as part of a two or three-dimensional view of the battlefield as opposed to raw data. Conclusions are then communicated based on the analytical results (Smith et al., 2012). If the answer to the question has not been satisfied or if further investigation into the problem is desired, the process is then repeated (see Figure 3).

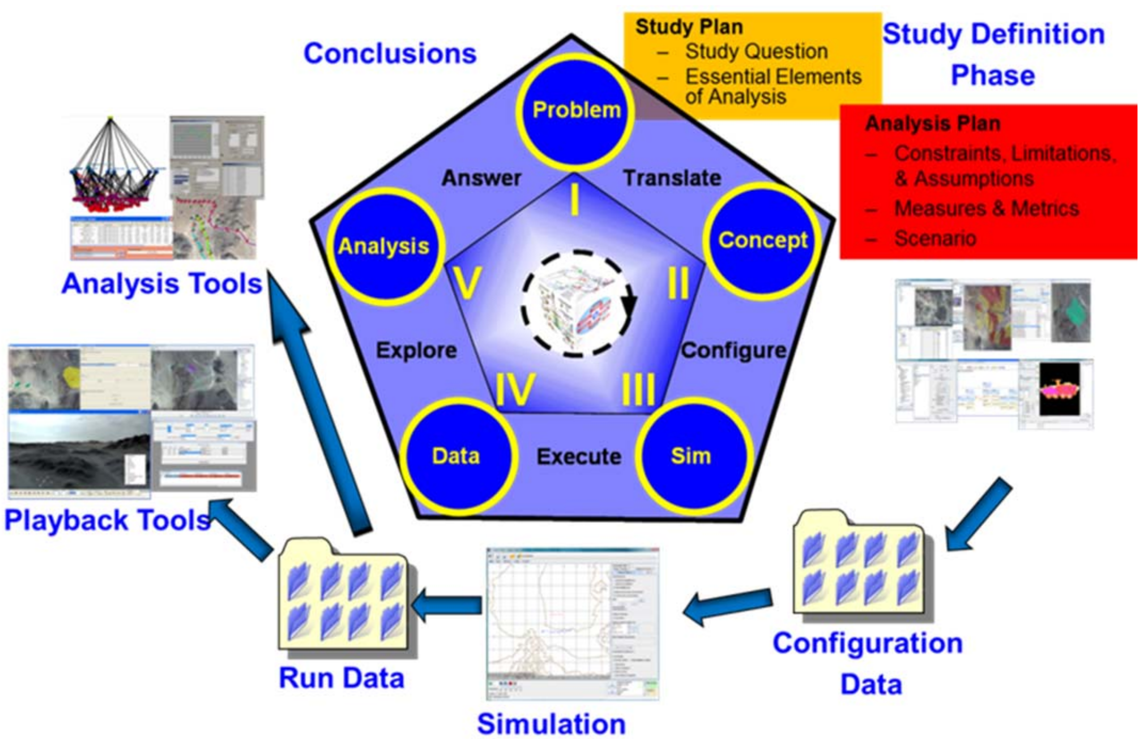


Figure 3. Traditional system of systems analysis process (from Smith et al., 2012).

E. OUTPUT AND ANALYTICAL TOOLS

The output for S4 comes in the form of traditional data files or playback data files. Traditional data files come in the form of raw data for the complete experiment, whereas playback data files are files of each individual simulation run that are used to visualize the battlefield during a certain instance. The analytical tools used in conjunction with the S4 model are generally commercial off-the-shelf analytical tools that are used to analyze multiple runs. For playback results, the results are displayed using NMSU-PSL's QuickLook for a two-dimensional view and another internally developed tool for a three-dimensional look. The limitation of the two playback options is that they can only look at one realization of a simulation run, but used in conjunction with more conventional analytical tools, the distribution of the occurrence of an interesting event can be determined.

F. SCENARIO

The scenario for the simulation is chosen to represent an urban area to stress communications among the agents in the model. The mission is to conduct a presence patrol in the three-square-mile urban area. The objective of the mission is to spot and communicate enemy locations and direction. In order to get the most out of the simulation, there is no engagement of the enemy when sighted. Additionally, agents within the model have perfect sensing. This allows for the simulation to really focus on the communications network and the success or failures of the network. The DMPs used are primarily those of communication with movement being the secondary DMP.

The agents in the model represent both a blue force and a red force. The blue force has the more complex DMPs, but they are still simple enough to be able to evaluate the communications network. The blue force structure consists of a BN CDR, one CO with a CO CDR as a key leader, and three PLTs. Each PLT has a PLT LDR and three SQDs, each with a SQD LDR and two TM LDRs. Additionally, there are four team members per team. Altogether, there are 86 blue force entities in the scenario (see Table 2). The red force has no objective and do not communicate with one another. They simply drive around in the urban areas. There is no command structure for the red force.

Once a red agent has been sensed, the sensing agent communicates the location and direction of the red agent to the other agents within its communications network. If an agent becomes aware of an enemy through organic sensing or through intelligence from the communications, it maintains awareness of that enemy unit until the end of the simulation.

Table 2. BLUE force agents.

AGENT	DESCRIPTION
BN CDR	Provides C2 to the blue force using the battalion DMP
CO CDR	Provides C2 to the blue force platoons using the company DMP
PLT LDR-X	Provides C2 to the respective platoon using the platoon leader DMP; there are three platoons, X={1,2,3}
SQD LDR-X/Y	Dismounted patrol squad leader; there are three squads per platoon, Y={1,2,3}, i.e., squad leader two of the third platoon is SQD LDR-3/2
TM LDR-X/Y/Z	Dismounted patrol team leader; there are two teams per squad, Z={1,2}, i.e., team leader two of third squad in the second platoon is TM LDR-2/3/2

The parameters used in the model are chosen by NMSU-PSL and ARL-SLAD to see what effect they have on communications within the model. NMSU-PSL and ARL-SLAD define the parameter space for each of the factors based on what they believe to be an acceptable range of values. In the study, there are currently 12 communications parameters used as inputs to the model. Two of the input parameters are discrete factors, eight are continuous factors, and the remaining two are categorical. Originally, there were 16 factors, but after a visit to NMSU-PSL, it was determined that two of the factors would be better served as output measures. One of the factors that was changed to an output measure is the data rate mode. The data rate mode is the rate at which data is transferred within the communications network. If the data rate mode starts at a certain rate, it can never change to a higher rate. Since the data rate mode changes depending on the demands of the network, changing the data rate mode to a response to see which mode the simulation uses most often seems to be more logical. The second factor changed to an output measure is the voice quality. The reason for this change is that the voice quality of communications transmissions may be affected by the other factors in the scenario. Another input parameter was divided into two separate inputs, and a categorical

variable was introduced bringing the total back up to 16. Issues with coding of the model later led to the reduction of two discrete factors and the combination of three continuous factors into one categorical factor leaving the scenario in its current state. A list of the parameters used in the scenario can be found in Table 3.

Table 3. List of parameters by type, range, and definition.

PARAMETER	TYPE	PARAMETER RANGE	DEFINITION
Network Configuration	Categorical	Configuration 1-3	Refers to the configuration of the communications network. This is the set up of communications equipment throughout the force
Urban Profile	Categorical	Profile 1-3	The profile of the urban area that defines the mean building height, mean road width, and maximum building seperation
Retries for Message Parts	Discrete	1-5 attempts	The number of times an attempt is made to communicate a message
Red Hilux Truck	Discrete	3-9 trucks	The number of enemy units on the battlefield
Maximum RR Transmit Power	Continuous	1-5 watts	The maximum rifleman radio transmit power is used to limit the communication distance within the communications network in which the radio belongs
Maximum MP Transmit Power	Continuous	1-20 watts	The maximum manpack radio transmit power is used to limit the communication distance within the communications network in which the radio belongs
Manpack Radio Trans Gain	Continuous	0-3 dB	Limits the ability of the manpack radio to receive messages
Rifleman Radio Trans Gain	Continuous	0-3 dB	Limits the ability of the rifleman radio to receive messages
Noise Power	Continuous	-180 to -90 dBm	Outside disturbances to the communications network
Antenna Height	Continuous	1-3 m	The distance from the radio to the end of the antenna plus a constant used to represent the height of the operator
Organic Voice Distance	Continuous	25-50 m	The distance that agents can communicate without the use of communications equipment
Relative Permittivities	Continuous	4-7.5 (unitless)	A measurment of how well communications are transmitted through different building materials

The network configuration parameter gives the scenario three different variations of the communications network. The three network configurations vary slightly to see how communications are affected based on the number and type of units that are

included in each sub network. The first network configuration (Figure 4) allows for communication from the BN CDR to the CO CDR, from the CO CDR to the PLT LDRs, and from the PLT LDRs to their respective SQD LDRs. Team members can only communicate within their own SQD. In order to communicate to another SQD within their PLT, information has to be sent through their SQD LDR. The second configuration (Figure 5) allows SQDs to intercommunicate within their own PLT. To communicate outside their PLT, information has to be passed through their PLT LDR. The last configuration (Figure 6) allows all entities within the CO to intercommunicate.

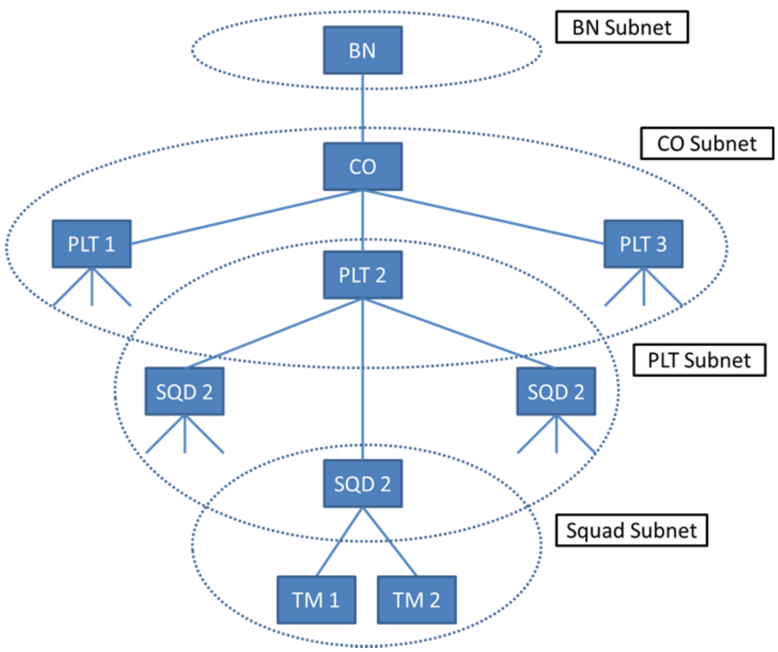


Figure 4. Network configuration 1.

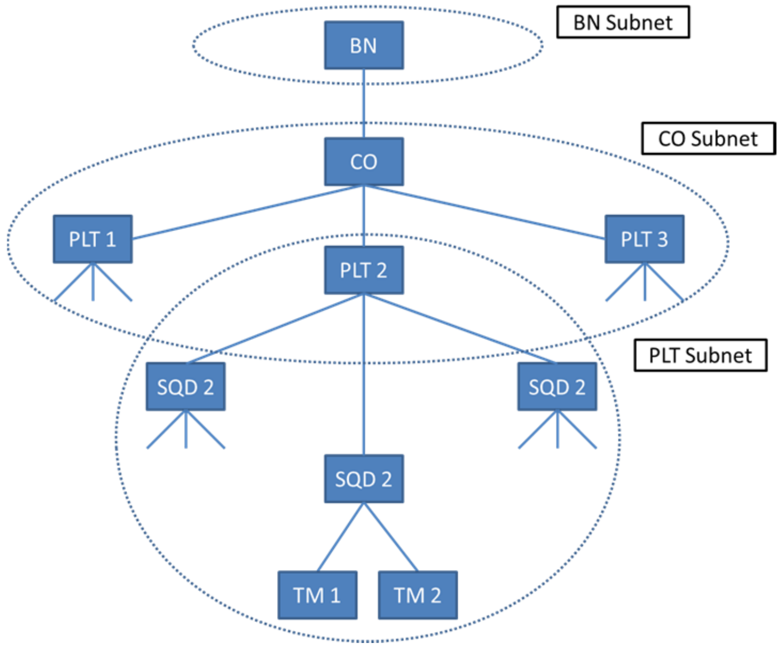


Figure 5. Network configuration 2.

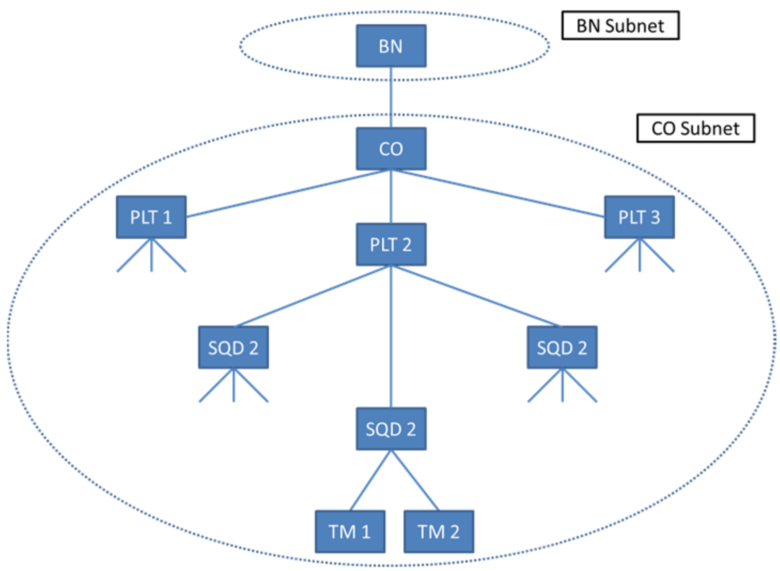


Figure 6. Network configuration 3.

The output measures are primarily based on blue situational awareness of red (BSAR) and blue situational awareness of blue (BSAB). Altogether there are eight response metrics for the scenario.

- Do all blue forces become aware of all sighted red forces?
- How long does it take for all blue forces to become aware of a red unit after it is first sighted?
- Are all red forces sighted?
- How much time does it take for blue forces to reach a destination point when impeded by a lack of communications? A blue force unit may search an area that has already been searched if they do not receive all of the updated locations of the other units.
- What is the message completion rate or latency?
- What is the minimum number of hops for communication to reach other nodes or agents (network connectivity)?
- What is the data rate mode (2000 kbps, 936 kbps, 112 kbps, 56 kbps) measure selected to transmit data messages?
- What is the voice quality (VQ) of messages received? VQ is required to declare a voice communications success. The default requires a VQ greater than 2.5 for success.

At first glance, some of the output measures appear that they may be correlated, if not highly correlated. There are 12 base case runs for the simulation in which the first four of the output metric results are provided. The output results for the amount of time that it takes the blue forces to reach their destination are broken down by PLTs. The average time that it takes a PLT to reach the destination is used for the base case analysis. The output results for time that an enemy unit is first spotted and the number of blue forces that are aware of the red forces is broken down by each of the enemy trucks. For the base case analysis, the time that it takes for blue forces to spot the last red truck is used. Also, the minimum number of blue forces that are aware of any of the red trucks is used. The number of enemy trucks used in the base case simulation is six. The perceived count of enemy units is a constant from zero to six. The base case results provided show that the blue forces are aware of all six enemy trucks in 10 of the 12 runs and they are aware of only five in the other two runs. These results are used to determine the correlation among the response metrics. The amount of time it takes for the last enemy unit to be spotted is highly correlated with the perceived number of enemy units in the simulation, as shown in Figure 7. The only metrics that do not appear to be correlated are

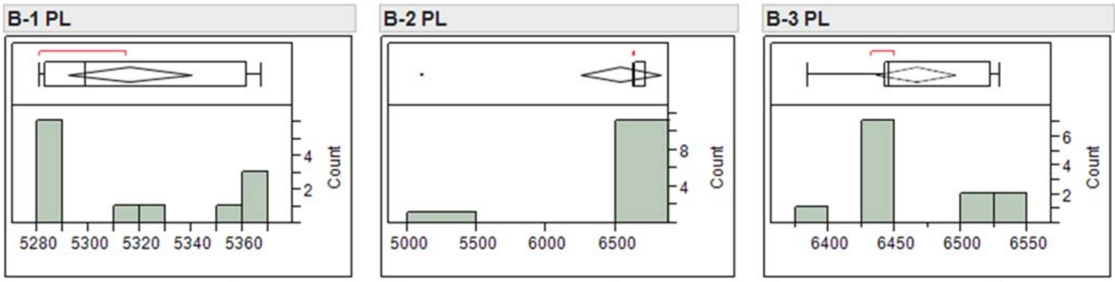
the time it takes for the blue forces to reach the destination and the number of blue forces that are aware of the red trucks.

	AVG_TimeToGoal	RED_PerceivedCount	MAX_FirstSpotted	MIN_NumBlueAwareOfRed
AVG_TimeToGoal	1.0000	-0.2824	0.3198	-0.0043
RED_PerceivedCount	-0.2824	1.0000	-0.9904	0.5199
MAX_FirstSpotted	0.3198	-0.9904	1.0000	-0.5295
MIN_NumBlueAwareOfRed	-0.0043	0.5199	-0.5295	1.0000

Figure 7. The correlation matrix for the response metrics show that the outputs of the simulation are correlated.

Results from the base case simulation are summarized for the amount of time it takes for the BLUE forces to reach their destination. The summary statistics for each of the three PLTs are shown in Figure 8. Given that there are only 12 base case runs and one of the runs produces an outlier, the variability of the second PLT may be inflated. This is important because the number of repetitions required for each design point in the DOE is influenced by the variability of base case simulation.

The statistical software JMP is used to provide visual and numerical descriptive statistics in Figure 8. JMP places the observations for each of the PLTs into a bin and provides a histogram and boxplot of the continuous data. The box-and-whisker plots above the histograms show the variation of the data based on the quantiles. It is evident by the visual display of the data that one observation from the second PLT is an outlier. The standard deviation in the summary statistics at the bottom of the figure for the second PLT is significantly different than that of the other two PLTs.



B-1 PL

Quantiles

100.0%	maximum	5367.5
99.5%		5367.5
97.5%		5367.5
90.0%		5366.3
75.0%	quartile	5361.63
50.0%	median	5299
25.0%	quartile	5283
10.0%		5281.6
2.5%		5281
0.5%		5281
0.0%	minimum	5281

Summary Statistics

Mean	5315.8333
Std Dev	37.619104
Std Err Mean	10.8597
Upper 95% Mean	5339.7354
Lower 95% Mean	5291.9313
N	12

B-2 PL

Quantiles

100.0%	maximum	6721
99.5%		6721
97.5%		6721
90.0%		6719.8
75.0%	quartile	6713.5
50.0%	median	6636.75
25.0%	quartile	6635.75
10.0%		5559.3
2.5%		5098.5
0.5%		5098.5
0.0%	minimum	5098.5

Summary Statistics

Mean	6534.625
Std Dev	453.8811
Std Err Mean	131.02419
Upper 95% Mean	6823.0073
Lower 95% Mean	6246.2427
N	12

B-3 PL

Quantiles

100.0%	maximum	6529.5
99.5%		6529.5
97.5%		6529.5
90.0%		6528.15
75.0%	quartile	6522.5
50.0%	median	6445.25
25.0%	quartile	6442.63
10.0%		6398.9
2.5%		6384.5
0.5%		6384.5
0.0%	minimum	6384.5

Summary Statistics

Mean	6465.5833
Std Dev	46.867722
Std Err Mean	13.529546
Upper 95% Mean	6495.3617
Lower 95% Mean	6435.805
N	12

Figure 8. Histograms and summary statistics for the amount of time it takes for BLUE forces to reach the destination.

THIS PAGE INTENTIONALLY LEFT BLANK

III. EXPERIMENTAL DESIGN AND FACTOR SCREENING

The S4 model has many parameters that have a wide range of values that are feasible. In order to properly identify the influential factors in the S4 model, the factors must be varied throughout their possible ranges. This leads to the development of designs to efficiently investigate the parameters and the range of values of the parameters, known as the parameter space. The first step in developing a design for the experiment is to identify and define the parameters of interest. Once the parameters are identified, the design is created to efficiently sample from the feasible space of the parameters and then the scenario is simulated through multiple runs of each design point within the design. After the completion of the simulation, the output can be analyzed to look for trends or insights from the defined parameter space. Additionally, output that is considered to be an anomaly can be further investigated to see what caused the abnormal behavior of the model. Within this research, multiple designs are created for use by the S4 team. The designs that are created are not implemented as part of this research, but they are available for later use by ARL-SLAD and NMSU-PSL.

A. DESIGN OF EXPERIMENT SELECTION

Design of experiment (DOE) has been in use for many years. Efficient designs may significantly reduce the time to run a simulation, while at the same time providing more detailed insights into a model's behavior. There are many designs that can be used for a simulation; each has its strengths and weaknesses. First, there is a 2^n full factorial design in which there are n factors that are observed at all possible combinations of their extreme settings one at a time. This can be useful when there are not many factors and when the response is expected to be linear in the parameters. However, when the response is not linear, there is no way to tell what is happening in the interior of the parameter space. This motivates us to add additional sampling in the design space. This can be done by including additional levels for each factor and creating an m^n factorial design, where m is the number of levels for each factor. Adding the additional levels increases the space-filling properties of the design and allows us to fit more complicated

meta-models, but at the cost of runtime of the experiment. Because of the amount of time that it takes for a simulation experiment to complete with the current resource capabilities, using these designs usually limits the experiments to just a few factors with a low number of levels (Sanchez & Wan, 2009). Since many models used today are complex, and contain hundreds or even thousands of variables that can be varied, smarter and more efficient designs must be used to explore these models. The S4 model itself contains over 300 input variables, and while only 12 are being initially investigated, this is more than enough variables to warrant the use of more efficient designs. While there are many methods for creating designs, this research is investigating the utility of nearly orthogonal Latin hypercubes (NOLHs) and nearly orthogonal, nearly balanced mixed designs (NONBMDs) that are more space-filling and require less design points than a full factorial design for analysis of the model. Additionally, a factor screening method that may allow the analyst to narrow the number of factors to be investigated prior to using one of the space-filling designs is explored to determine its potential usability for the S4 model as well as other models.

The design selection is influenced by the run-time required for the DOE. Upon receiving the base case runs for the S4 model, the response metrics are analyzed to determine the total number of replications needed to ensure that confidence intervals with the desired confidence coefficient and precision can be generated. The response metric for the amount of time needed for BLUE forces to make it to a destination point in minutes is used to determine the number of replications. The initial base case run of the model includes only 12 replications of the one design point. Because of the low number of initial runs, the variation of the response may not be accurately estimated. That being said, the PLT with the highest variability is used to calculate the number of replications needed to achieve a statistical resolution of being within 60 minutes from the mean with a confidence-level of 95%. The calculation results in requiring 491 replications for each DP within the simulation. The other two PLTs require only four replications to achieve the desired power due to their significantly smaller standard deviations. This is an important factor in selecting the design, but the number of replications required may be reduced if the number of base case replications is increased in order to give a more

accurate estimate of the variation or less statistical power is required. The sample size calculation is achieved by the following formula where n is the replication size, σ is the standard deviation of the base case, z_α and z_β are the standard scores for the levels of confidence and power respectively, and $\mu_o - \mu_\alpha$ represents the desired statistical resolution (Devore, 2011):

$$n = \left(\frac{\sigma * \left(z_\alpha + z_\beta \right)}{\mu_o - \mu_\alpha} \right)^2$$

The graph in Figure 9 shows the number of replications necessary to be within a given amount of time from the mean with a 95% CI while maintaining certain levels of power. The same graph is created for the same PLT with the removal of a single data point considered to be an outlier (Figure 10) and once again for an alternate PLT (Figure 11). The power calculations in Figure 10 and Figure 11 are very similar, which leads to the assumption that the variation of the base case data is causing the replication calculation to be erroneously inflated. Additional base case runs are needed to be able to accurately determine the replication size needed for the simulation.

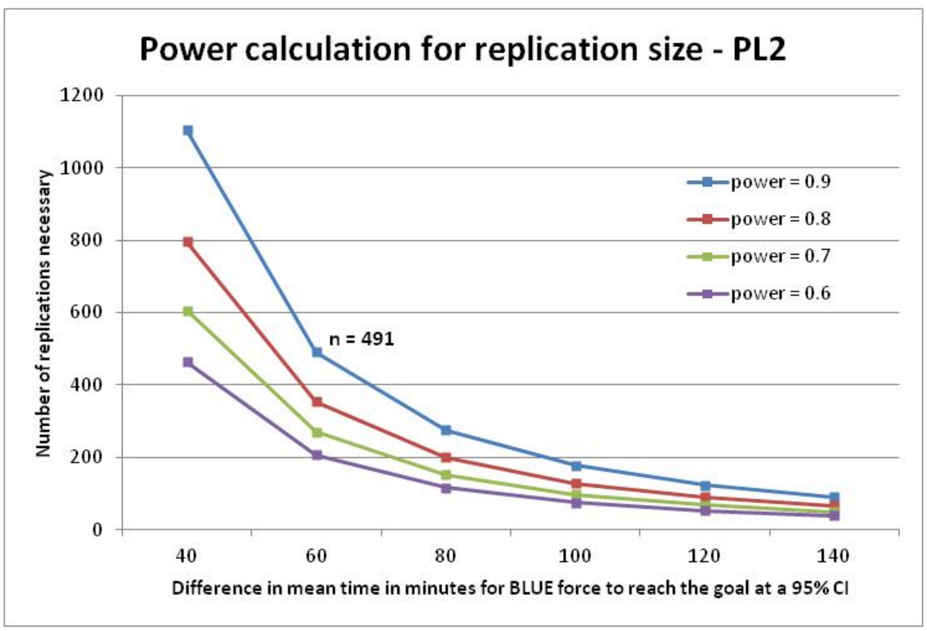


Figure 9. Power calculation for replication size needed with PLT two from the base case data.

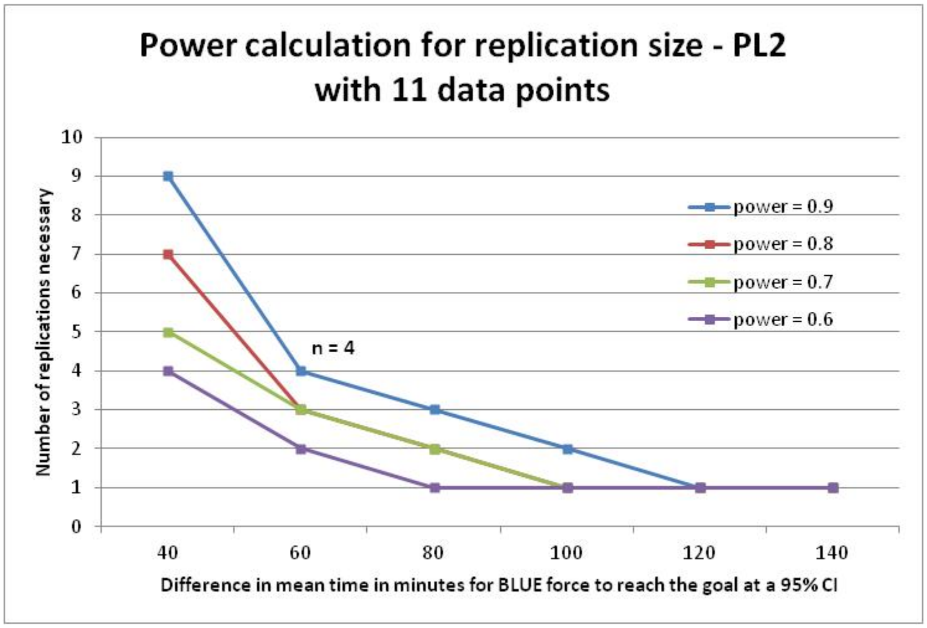


Figure 10. Power calculation for replication size needed with PLT two from the base case after removing one data point.

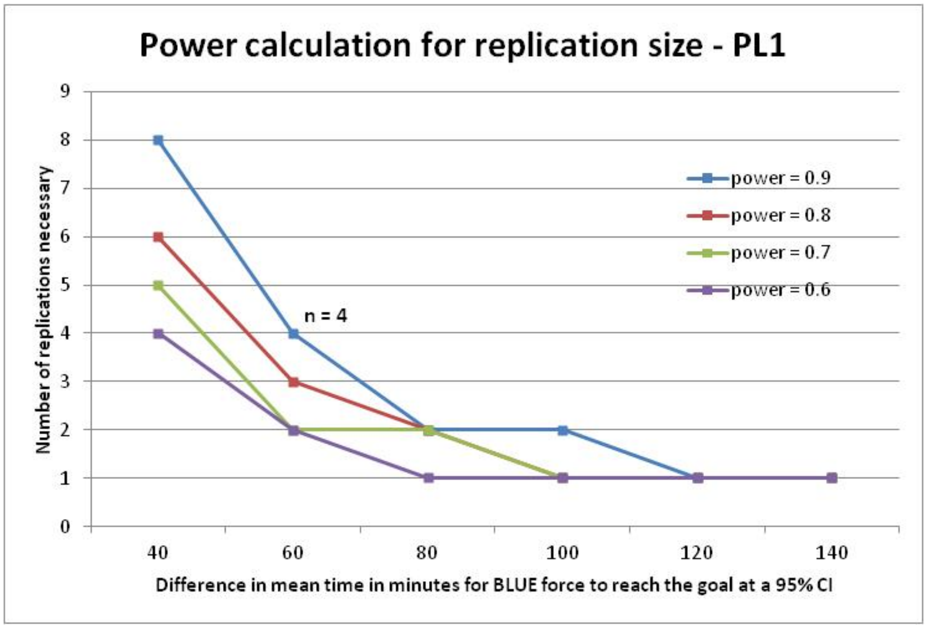


Figure 11. Power calculation for replication size needed with PLT one from the base case.

1. Nearly Orthogonal Latin Hypercube

The Nearly Orthogonal Latin Hypercube (NOLH) designs developed by Cioppa and Lucas (2007) allow for designs to be quickly created for up to 29 factors. If there are more factors, the methods of Hernandez et al. (2012) can be used. These designs are space-filling and efficient. One of the benefits of the NOLH designs is that the number of design points needed is reduced as compared to full factorial designs. One drawback of the NOLH designs is that they are intended for use only with continuous factors. The NOLH can still be used with both discrete and categorical factors, but the performance of the maximum absolute pairwise correlation cannot be guaranteed.

With only 12 factors being varied, it would be possible to use one of the NOLH designs that only has 65 design points, but because the NOLH designs were created for use with continuous variables and the S4 model parameters also include categorical and discrete variables, a 129-design point matrix is used to offset some of the rounding errors that are created when using the discrete variables. Taking away the categorical variables to have the design run for each combination of the variable levels produces a maximum absolute pairwise correlation for the rest of the design columns of 13.3%, which is too high to be considered "nearly orthogonal." This 129-design point NOLH design matrix is portrayed in Figure 22 located in Appendix A. The design is created by using the NOLHdesigns spreadsheet (Sanchez, 2011)

Another idea is to use the 129-design point NOLH without the categorical variables, network configuration and urban profile, and run it for each combination of the categorical variables, three levels each (for a total of nine combinations). This design produces a maximum absolute pairwise correlation of only 4.66%, and is thus considered to be nearly orthogonal, but at the expense of increasing the number of design points from 129 to 1,161. This leads to the desire to find a better way of dealing with both the discrete and categorical variables since each design point would need to be run 491 times based on the sample size calculations from the base case analysis.

2. Nearly Orthogonal Nearly Balanced Mixed Design

The Nearly Orthogonal Nearly Balanced Mixed Design (NONBMD), just like the NOLH, attempts to minimize the maximum absolute pairwise correlation between the design matrix columns, but also takes into account the presence of discrete and categorical variables and the imbalance that they may cause (Vieira et al., 2012). By using the design spreadsheet by Vieira (2012), a 512-design point NONBMD is created in which the maximum absolute pairwise correlation is less than three percent (see Figure 23, Appendix A). This design requires around half of the runtime of the NOLH and has a lower maximum absolute pairwise correlation.

3. Design Creator

The original idea for a design for the S4 model was to use the design creator spreadsheet from the dissertation of LTC Alex McCalman (2012) to create a second-order NONBMD that would ensure the maximum absolute pairwise correlation from the design is below a certain threshold for not only the factors but also their quadratic and interaction terms. With the number of factors in the design, the design creator takes days to run. Each time it was run, it failed to complete within the correlation threshold which terminated the program. Additional attempts could have been made, but the number of design points needed would be too many to complete the simulation in a timely manner.

Instead, a 33-point design using the same design creator (McCalman, 2012) is created using only first-order effects for the design (see Figure 24, Appendix A). The design is created only for the discrete and continuous parameters with the idea of using a cross-design in which the same design is used for all of the combinations of the two categorical parameters. This design remains space-filling and allows the model to be explored while only using 297 total design points and keeping the maximum absolute pairwise correlation at less than 1%, improving on both runtime and correlation as compared to the original NONBMD. The downfall to using this design is that the space-filling properties are not as good as the other two and there are fewer samples taken from the corners of the parameter space.

B. FACTOR SCREENING

Sometimes there is insufficient time to both experiment with designs and analyze results of a simulation, especially in large-scale simulation models where there could be many influential factors. A factor screening process can be used to explore the multiple factors across the value space by creating a supersaturated design (SSD) for experimentation and using variable selection methods to analyze the results, thereby reducing the amount of time it takes to run a full blown experimental investigation. A supersaturated design is a design that has more factors than design points. Generally, the Lasso method and other L_1 penalty based variable selection methods are used as the approach to the analysis step of the investigation (Xing Wan, Zhu, Sanchez, & Kaymal, 2013). The Lasso method is a type of least squares regression analysis in which a penalty is applied to the set of factors that causes many of the coefficients of the insignificant factors to shrink to zero prior to selecting the important factors. The methods and criteria of this approach are fairly complicated mathematically, so the approach of this thesis is to see how effective a stepwise regression technique will be in identifying the key factors. The effectiveness is based on the probability of correctly identifying the influential factors and properly determining whether the sign of the coefficients associated with the factors is positive or negative.

Coding issues with the S4 model did not allow enough time for ARL-SLAD and NMSU-PSL to complete the S4 simulation with any of the DOEs previously discussed. This factor screening technique is introduced as an alternative approach for identifying key factors within the S4 model. Additional work will be required before the approach can be used with S4, as an optimal SSD is required to be constructed based on a specific design structure criteria.

C. FACTOR SCREENING MODEL DEVELOPMENT

In order to test the capability of using a stepwise regression factor screening technique to properly identify key factors in a model, the model must be developed and tested on models for which the influential factors are already known. The idea behind this is to create a linear regression model, $Y = X\beta + \varepsilon$, where $X = (x_1, x_2, ..., x_p)$ is an n x p

supersaturated design matrix and each x_i is a vector of length n, $Y = (y_1, y_2, ..., y_n)'$ is the response vector, $\beta = (\beta_1, \beta_2, ..., \beta_p)'$ is the vector of regression coefficients, and $\varepsilon = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_n)'$ is the error. The plan is to simulate the linear regression model from a model where the β vector and the design, X, are known and are used to generate the random Y response vector. Stepwise linear regression is then used on the design matrix with the response to determine the significant factors.

A DOE on the components of the stepwise regression factor screening technique is used to test how well the factor screening technique works. One reason for this experiment is to test the sensitivity of the ability of the technique to properly identify significant factors for various real-world models. Another reason for the experiment is to be able to compare results of the stepwise linear regression with the Lasso method in future research.

Conducting the experiment where the significant factors are known ahead of time allows the results of the stepwise regression to be compared to the known significant factors. The simulated responses for the stepwise regression are directly compared to the initial β values to see if the selection of the stepwise regression corresponds to a non-zero β value. If the selection does correspond to a non-zero β value, then the stepwise regression properly identifies the significant factor, otherwise the selection is incorrect. The experiment is conducted to determine the ability of the factor screening technique to properly identify the significant factors as well as the sign of the selected significant factors.

Altogether there are four components for the model used in the DOE: the number of significant factors, m, the mean of the regression coefficients, μ, the number of steps for the stepwise regression to use, s, and the standard deviation for the random noise, σ. The first m regression coefficients are randomly generated from a normal distribution with mean μ and standard deviation of one, $N(\mu,1)$, and the noise, ε, is randomly generated from a normal distribution with a mean of zero and a standard deviation of σ,

$N(0,\sigma)$. Model output is simulated from all combinations of the components, for a total of 1944 combinations. The range of the parameters is listed in Table 4.

The basic flow of the code for the model is as follows:

1. Initialization of the β matrix: create an $n \times p$ matrix of zeros, where n is the sample size for the experiment and p is the total number of factors.

2. Define the first m significant factors of each row in the β matrix, $\beta_i \sim N(\mu,1)$ for i in 1 to m.

3. Take the product of the first row of the β matrix with the supersaturated design, X, to create the response vector, Y, then add random noise.

4. Combine the X matrix with the Y vector and use stepwise regression to select significant main effect factors and then create a row of calculated coefficients for all of the factors in the model.

5. The output stored for each sample is the original row from the β matrix combined with the row of calculated coefficients as a single row of numbers.

6. Repeat the process for each row in the β matrix.

7. Use the stored output for all of the samples to create a single matrix and then multiply the values of the first m columns of the matrix with the first m columns following column p to create an $n \times m$ matrix to be exported as a comma separated value (CSV) file.

Table 4. Factor screening model components.

PARAMETER	PARAMETER RANGE	DEFINITION
m	2:10	The number of significant factors
μ	0:5	The mean of the β values
s	m:$m+8$	The number of steps used for the stepwise regression
σ	2:5	The standard deviation of the random noise

A supersaturated matrix with 24 design points and 69 factors is used in the model as our design matrix, X. Along with the design matrix, the regression coefficient vector, β, is created as stated before with the first m values randomly generated with a given

μ . These randomly generated β values are used as the significant factors. The rest of the β vector enteries are given a value of zero. Creating the β vector in this way allows all of the significant factors to be at the beginning of the vector. The β vector is then multiplied by the design matrix forming the response vector. Once the response vector is created, the randomly generated noise vector, ε, is added to the response vector. Stepwise regression is then used to determine the significant factors for the linear regression model. This process is replicated 10,000 times for each combination of the parameters.

The code for the model is written in the R programming language. The output of the code is in CSV format and each file contains 10,000 rows of data for a single combination of the model components. Altogether, there are 270 simulated data files used for the analysis of the factor screening technique. The simulation experiment was run constantly over 10 days using four core processors. Upon completion of the simulation, formulas were written in the CSV files for each of the response metrics analyzed.

IV. ANALYSIS

The analysis contained within this chapter pertains to the stepwise regression factor screening technique discussed in the previous chapter. The purpose of the analysis is to determine the probability of positively identifying all of the significant factors using a stepwise regression factor screening technique given a design and output responses. Additionally, the analysis shows the proportion of significant factors that are detected using the stepwise regression. Another item measured is the percentage of time that the significant factors are identified in the regression model with the wrong sign for the coefficients.

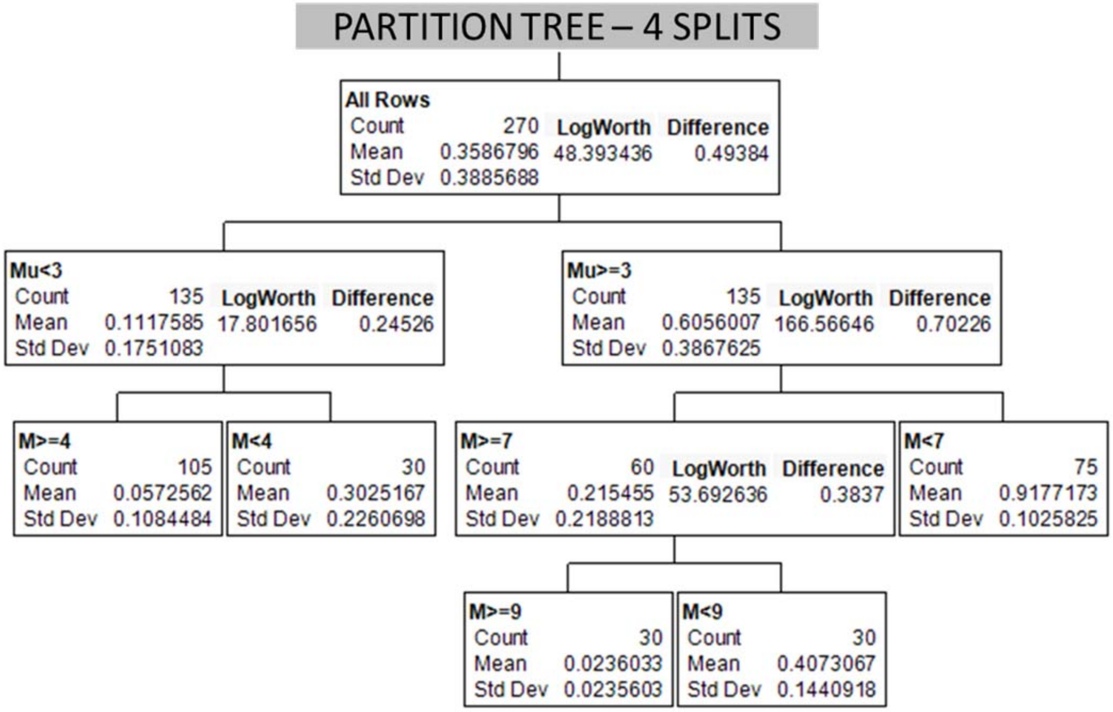
A. PROBABILITY OF DETECTING ALL SIGNIFICANT FACTORS

For the analysis of calculating the probability of selecting all of the significant factors using the stepwise regression, only three of the input components for the model are used: the number of true significant factors, the mean of the β coefficients, and the number of steps for the stepwise regression. A formula is written within the output of the model that counts the significant factors of the model if the coefficient does not equal zero, meaning that the factor was properly identified. An additional formula is written that counts the number of occurrences in which the previous count is equal to the number of significant factors, then divides this count by the total number of runs, which is 10,000. This results in the probability of correctly identifying all of the significant factors for the combination of the factors. This code is repeated for all combinations of the components.

The probability of correctly identifying all of the significant factors is analyzed for all of the combinations of components to determine what components appear to be important. One way of determining the important components of the model is by partitioning the resulting probabilities of detection of the model with respect to the model components. The partitions occur where the disparities in the data are largest. As the number of partitions of the data is increased, the importance of the split is decreased compared to the previous split. Since the partitioning occurs at the largest differences of

the data and the significance of the partitions decrease as the number of partitions is increased, the most important factors will be used to partition the data first (Gaudard et al., 2006). The two model components that are most important in the ability to detect all of the significant factors are the mean of the β coefficients and the number of significant factors. The number of steps does not increase the probability much after the first increase in the number of steps. This is evidenced by the partition tree in Figure 12, which shows that the partitioning occurs only with changes in μ and m. Continuing with the partitioning, even after 30 splits of the data, the number of steps is still not included as a partitioning factor. This further illustrates the relative insignificance of the number of steps in determining the probability of positively identifying all of the significant factors in the model.

The plots in Figure 13 and Figure 14 show the increase in the probability of detection of all of the significant factors as the mean of the β coefficients is increased. The probability is significantly decreased when the mean of the β coefficients is two or less. This may be explained by the introduction of the random noise that could counteract the random β values since the random noise has a mean of zero and a standard deviation of two, causing the significant factors to appear to be around the same significance as some of the unimportant factors. The plots in Figure 15 show how the probability of detection of all of the significant factors decreases as the number of significant factors is increased.



RSquare	RMSE	N	Number of Splits	AICc
0.899	0.1232611	270	4	-351.92

Figure 12. Partition tree with the probability of detecting all significant factors as the response. Partitioning initially occurs when $\mu \geq 3$ and $m \geq 7$. The “Count” refers to the number of instances in each partition and the “Mean” refers to the probability of detecting all significant parameters for the partition.

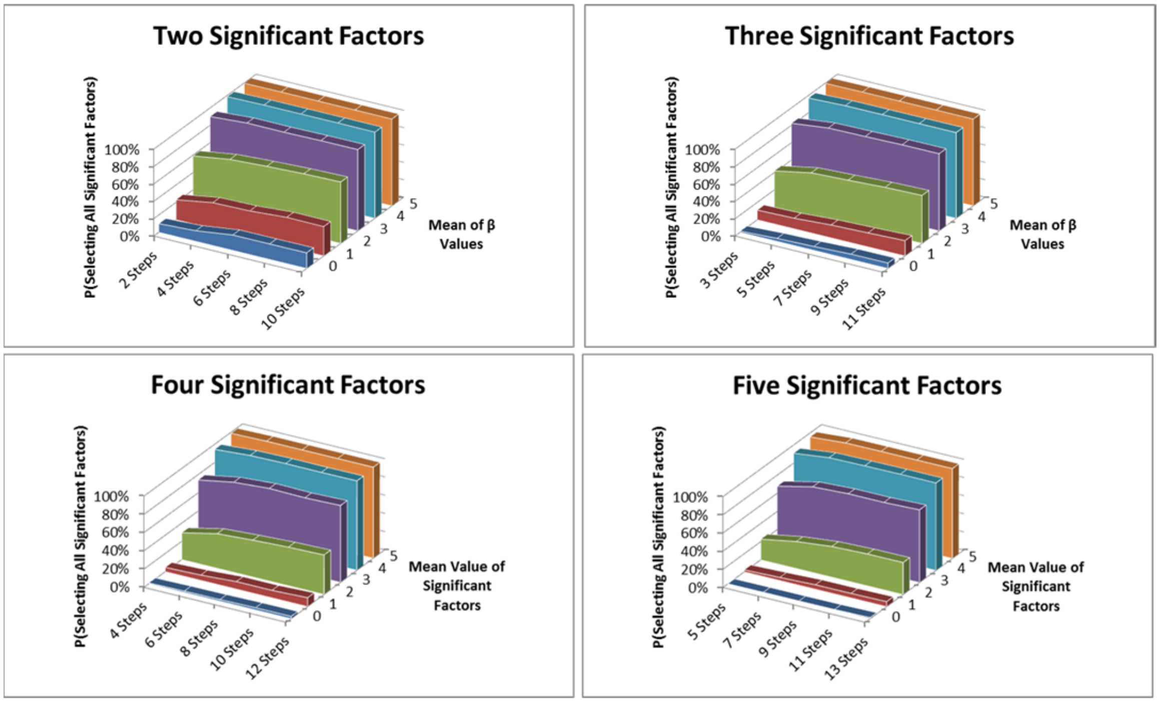


Figure 13. Three-dimensional plots of the probability of detecting all of the significant factors showing the significance of the mean value of β .

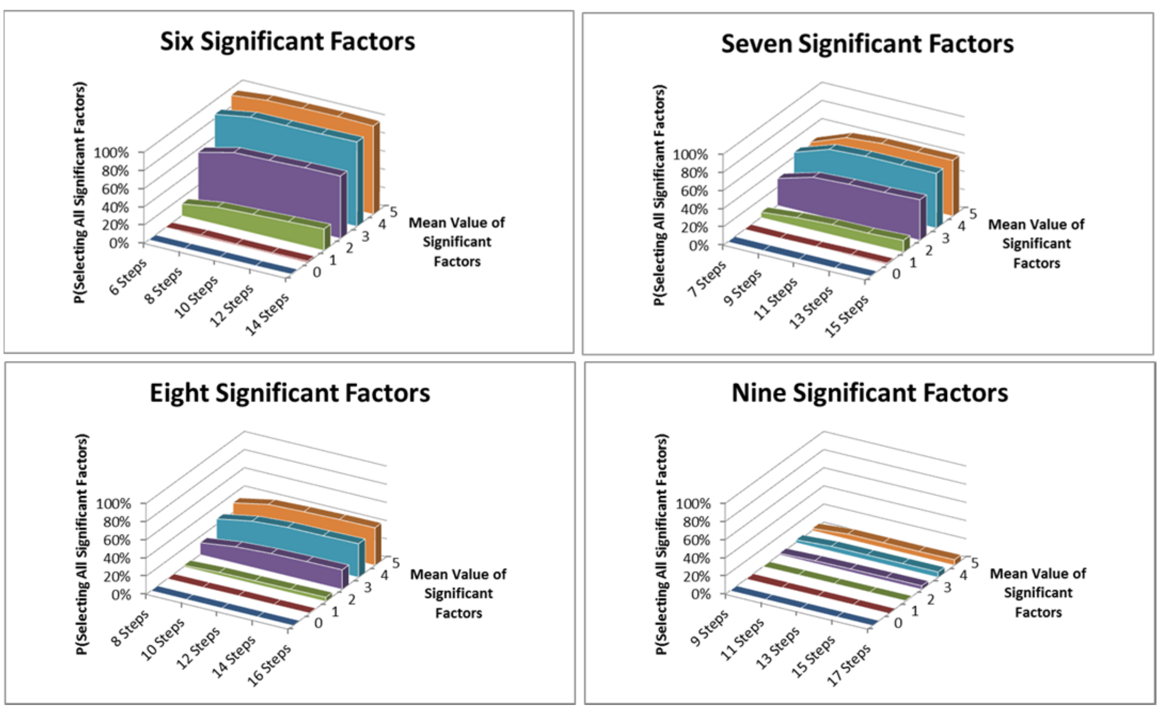


Figure 14. Three-dimensional plots of the probability of detecting all of the significant factors showing the significance of the number of factors.

Keeping in mind that there are 69 possible factors that potentially could be selected, the stepwise regression is capable of detecting all of the factors with a probability greater than 90% when the mean of the β values are greater than four and when the number of significant factors is six or less. As the number of significant factors is increased further, the success of the stepwise regression declines rapidly.

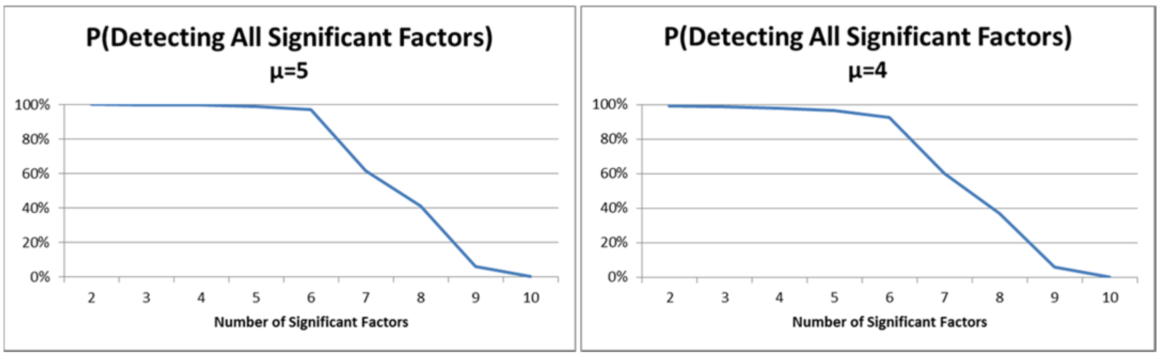


Figure 15. Plots showing the decrease in the probability of detecting all of the significant factors when the number of significant factors is greater than six.

As the mean of the β values draws closer to zero, the probability of detecting all of the significant factors also decreases. The stepwise regression performs moderately well when the number of significant factors is not greater than six and when the mean of the β values is three. Once the mean drops below three, the probability of detection begins to drastically decrease. This is consistent with the partitioning of the data previously discussed, and is further illustrated in the plot for the probability of detecting all factors shown in Figure 16.

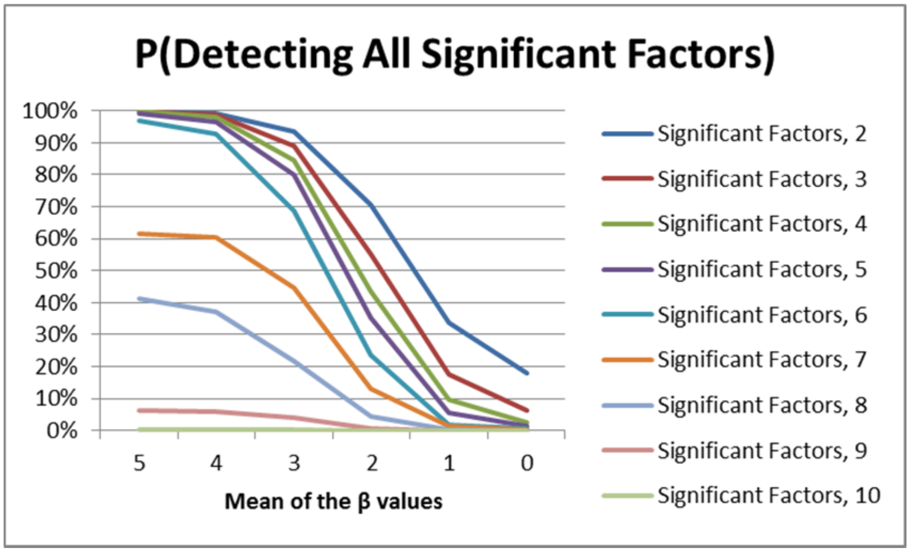


Figure 16. Plot showing the sharp decrease in the probability of detecting all of the significant factors.

B. PROPORTION OF SIGNIFICANT FACTORS DETECTED

As with the probability of detecting all of the significant factors, the proportion of significant factors that are selected are also affected mostly by the number of significant factors and the mean value of β. The number of steps still does not appear to be important. Partitioning of the data shows that the first split of the data occurs when $\mu \geq 2$. The partition tree in Figure 17 indicates that μ and m have the most influence in determining the proportion of significant factors detected. The number of steps, s, for the stepwise regression is more important in determining the proportion of significant factors detected compared to the probability of detecting all of the significant factors as it only takes 15 splits of the data before it becomes a partitioning factor, but it is still rather insignificant compared to the other two factors.

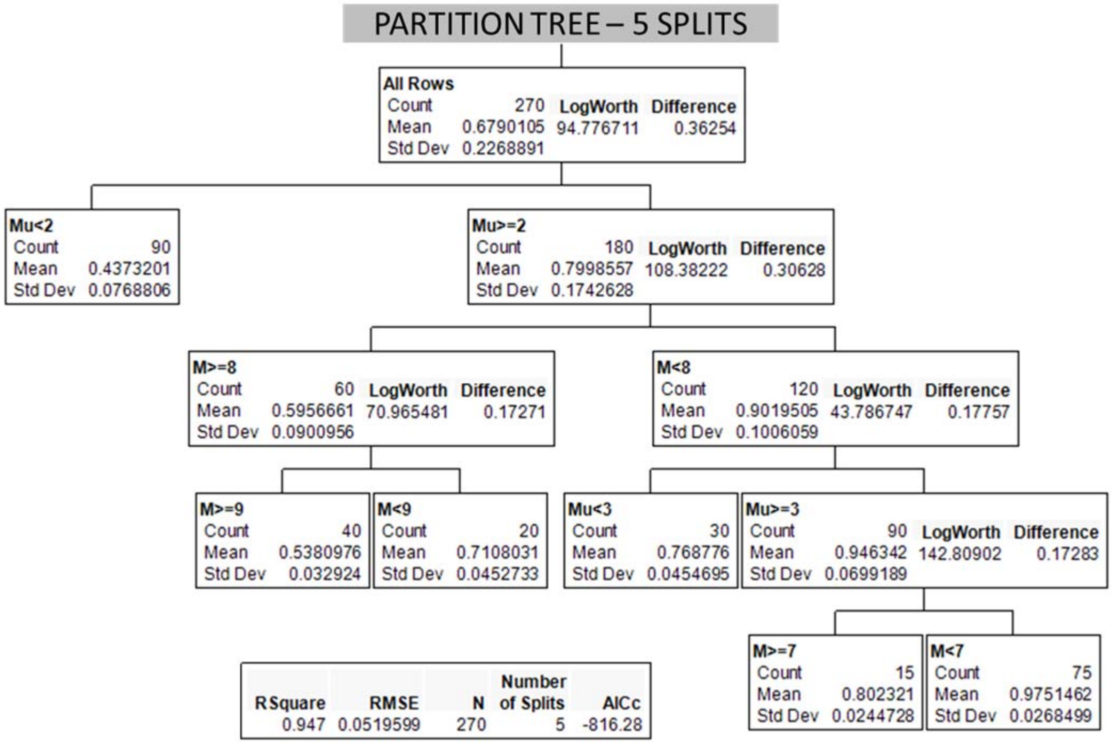


Figure 17. Partition tree with the proportion of significant factors detected as the response. The "Count" refers to the number of instances in each partition and the "Mean" refers to the proportion of significant factors detected for the partition. The tree shows that the majority of the influence on the response is from μ and m.

While both the mean value of β and the total number of significant factors are influential on the ability of the factor screening technique to properly identify the factors, when the number of significant factors is increased, the mean value of β becomes less important in the proportion of significant factors that are identified, as shown in Figures 18 through 20. The proportion of positively identified significant factors tends to level as the number of significant factors is increased for all β values.

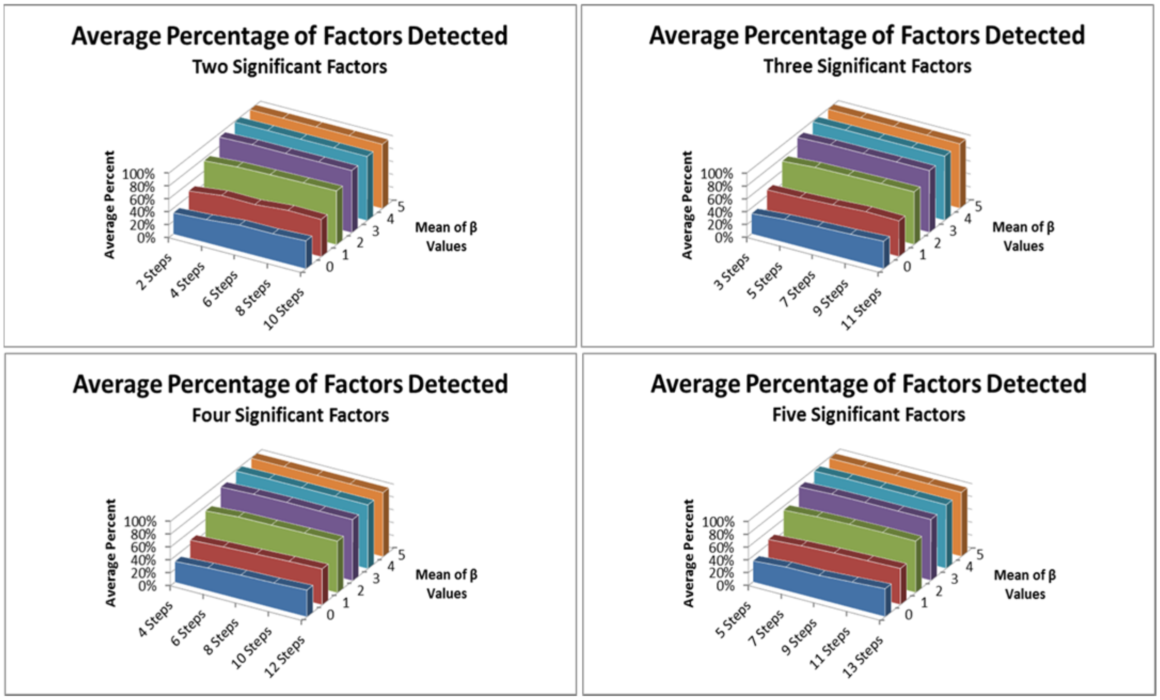


Figure 18. Plots showing the proportion of significant factors that are positively identified for 2 to 4 factors.

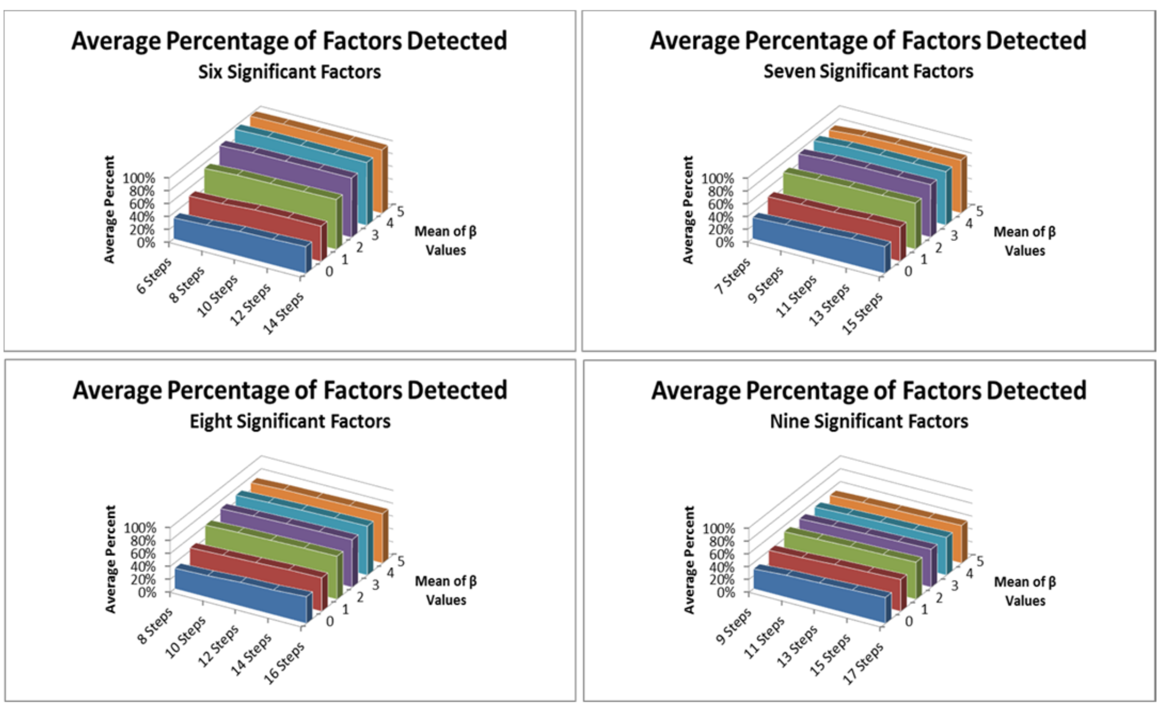


Figure 19. Plots showing the proportion of significant factors that are positively identified for 5 to 9 factors.

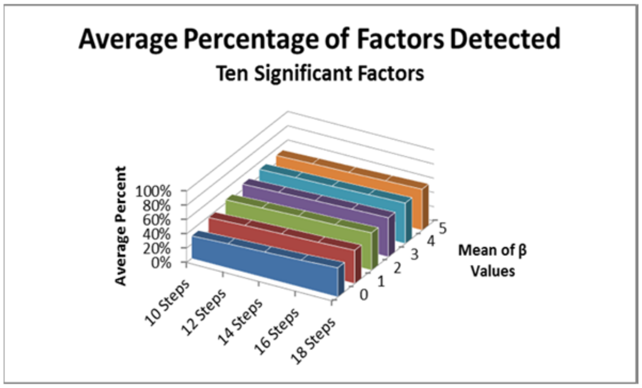


Figure 20. Plot showing the proportion of significant factors that are positively identified for 10 factors.

C. PROBABILITY OF INCORRECTLY ASSIGNING THE COEFFICIENT SIGN

Along with the proper selection of the significant factors, it is important that the appropriate sign for the coefficient of the significant factors is correctly identified. Within the output of the model, a formula is written to multiply the randomly assigned β values for the factors by the coefficients that the model returns for the selected factors. If the resulting product is negative, it indicates that the model incorrectly assigned the sign of the coefficient to the factor. The number of incorrectly assigned signs of the coefficients of the significant factors is then divided by the number of correctly identified significant factors for each replication. Averaging this ratio provides the probability of incorrectly assigning the coefficient sign for the significant factors.

Almost no false identification occurs when the mean of the β values is greater than one. This can more than likely be attributed to the fact that when $\mu = 2$ and even if the random noise is negative, the random noise that is added to the β values would generally cause the factor to appear insignificant. For $\mu > 2$, it would have even less effect on the classification of the sign of the coefficient. The probability of incorrectly assigning the sign of the coefficients for the significant factors increases as the number of significant factors increases. The plot in Figure 21 shows this increase for the number of significant values when $\mu = 0$. The number of steps used for the stepwise regression

does not appear to influence the probability of incorrectly assigning the sign of the coefficients for the significant factors.

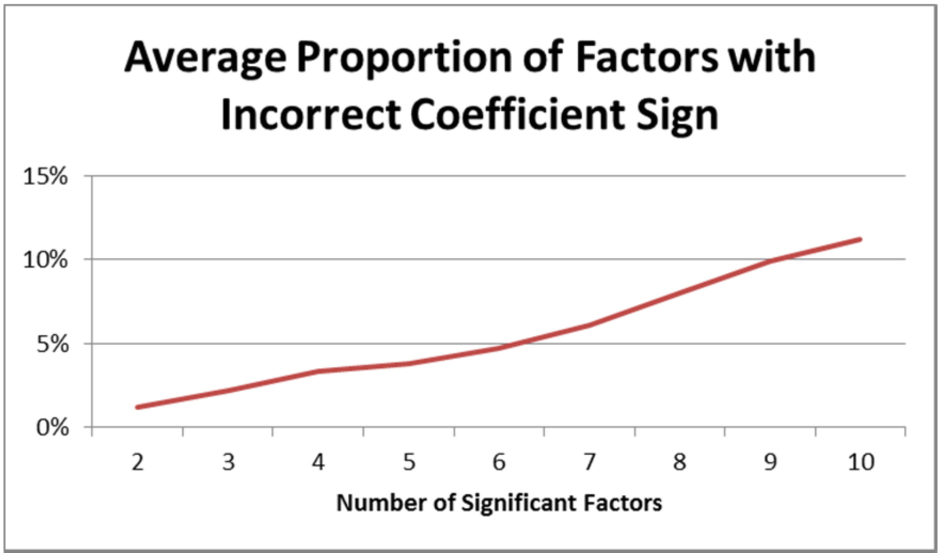


Figure 21. This plot shows the average proportion of factors with the incorrect sign of the coefficient based on the number of significant factors when $\mu = 0$. This indicates an increase in the probability of falsely identifying the sign of the coefficient as the total number of significant factors increases.

V. CONCLUSIONS AND RECOMMENDATIONS

The original direction of this thesis was to determine the most influential communications factors within the S4 model. A secondary goal of the thesis was to use different DOEs for the S4 model to see if there were any significant differences among the findings of the DOEs. Because of coding errors in the sponsor model that led to changes in the scenario and the model parameters to be used, there was not enough time to run the S4 model and analyze the results. The focus of the thesis changed more toward a theoretical approach of determining significant factors from a model using a factor screening technique that would allow the analyst to reduce the amount of time necessary to both run and analyze a model.

A. RESEARCH QUESTIONS

This purpose of this thesis was driven by the following questions:

1. What are the driving or most influential communications factors in the S4 model?
2. Given a supersaturated design (SSD) with a limited number of design points, can influential factors be properly identified using a stepwise regression factor screening technique?
3. How effective will a factor screening technique using stepwise regression be in identifying influential factors within the S4 model?

With these questions in mind, the following sections discuss the results of the analysis.

1. Influential Communications Factors

While the thesis set out to determine the most influential communications factors, time constraints disallowed the data from being produced by the S4 model and analyzed. This being said, ARL-SLAD and NMSU-PSL are now equipped with the capability of creating their own NOLH designs to use as they see fit with the S4 model. Additionally, a handful of designs were created specifically for the parameters of interest for the S4 model and distributed to the S4 team at NMSU-PSL. The factor screening technique used for the analysis of this thesis may also provide a way ahead for determining

influential factors for S4 as well as other models that have many input parameters and take considerable amounts of time to run.

2. Factor Screening Technique

The ability of the factor screening technique to identify significant factors using stepwise regression is dependent upon the total number of significant factors and their coefficients. When the coefficient of a factor is really low, the factor screening technique has a difficult time detecting it as being an important factor. Additionally, if the number of significant factors is more than six, the probability that all of the factors are selected using the stepwise regression begins to decline. That being said, the percentage of significant factors that are selected tends to be pretty good for settings where the factors are moderately to highly significant, and when the number of significant factors is lower. Overall, the technique is capable of identifying influential factors, but it does have its limitations. It could prove to be a useful method to use when there is not enough time to run a full DOE. Additionally, the factor screening method could be used as a preliminary selection method to be used in conjunction with an NOLH DOE.

3. Factor Screening Technique and the S4 Model

The effectiveness of using the factor screening technique cannot be directly measured as of yet, but the S4 model may be a good candidate for test runs of the technique. Previous analysis on significant factors in the S4 model resulted in the detection of only a couple of significant factors. Since the factor screening technique works well when there are few significant factors, the technique should be able to properly identify the factors. The main concern with the use of the factor screening technique within the realm of the S4 scenario described in this thesis is that the number of parameters that are being explored is few and the use of the technique may not necessarily save time compared to an NOLH DOE. Although it may not be useful for the given scenario, there are many parameters in the S4 model that have not been explored simultaneously. If all of the communications parameters within S4 are explored at one time, as opposed to only a select few, then the factor screening method may become more useful for identifying key communications factors. Additionally, the communications

model of S4 is just one small piece of the overall S4 model. When the S4 model is explored more in depth using the other sub-models, the factor screening method could be extremely useful in narrowing the number of parameters that may be significant for the desired response.

B. RECOMMENDATIONS FOR FUTURE STUDY

While this thesis set out to explore the communications model within S4, conditions unforeseen redirected the focus of the thesis toward parameter selection using factor screening and stepwise linear regression with a supersaturated design. This technique is still in the beginning stages and will need additional experimentation and analysis to see if it is truly a worthwhile method for factor selection. Future studies will be needed for both the S4 model and the factor screening technique, possibly using the technique with the S4 model for parameter selection.

1. DesignCreator for Second-Order Effects

The DesignCreator (MacCalman, 2012) is used to create a design for this thesis that allows for less run time than the other designs with lower absolute pairwise correlation, but it is capable of much more. The spreadsheet can be used to create a design that would allow second order effects to be explored while minimizing the correlation in the columns of the design. ARL-SLAD and NMSU-PSL expressed their desire for such a design to be used in S4 experimentation. Attempts were made to create a second order design with the given parameters, but were unsuccessful due to the number of design points required to achieve a design that has a maximum pairwise correlation within the desired threshold. Either further reduction in the number of input parameters or increased computational speed would be required to create a satisfactory design. The addition of such a design would be beneficial to the S4 team.

2. Addition of a Penalty for the Factor Screening Technique

While using the stepwise regression factor screening technique, the number of steps for the stepwise regression is given as a parameter. In all cases the regression would choose as many factors as the number of steps, regardless of how significant they were.

The addition of some type of penalty for adding unnecessary factors may reduce the number of factors selected thereby decreasing the run time for follow on designs. One issue with implementing a penalty is that it would be difficult to determine the appropriate penalty to use in the regression process. Exploratory experimentation and analysis may provide insights for accomplishing the addition of proper penalties and increase the accuracy of the factor screening technique.

3. Exploration of Non-linear Effects Using a Factor Screening Technique

One limitation of the stepwise regression factor screening technique using the supersaturated design is that it is only able to address main effects factors for a model. While this is useful, if a factor is not selected based on its influence as a main effects parameter and is not further explored, important interactions between the non-selected variable and other variables may be missed. Since the idea behind this factor screening technique is to use a supersaturated design in which there is a shortage in degrees of freedom, it may not be feasible to use this technique to explore non-linear effects. It may be possible, however, to reduce the number of factors that are being explored in order to include the second order effects within the supersaturated design. This would not allow the screening of large numbers of factors, but it may prove to be an alternative exploration of a model in which there are only a few factors of interest.

C. RESEARCH SUMMARY

This thesis began with a focus on the communications environment of the S4 model, but was altered to explore factor screening to support future S4 experiments and other model exploration. Factor screening using a supersaturated design could prove to be a valuable tool in that it can provide the capability to identify significant factors for a model with large numbers of factors with fewer design points than the total number of factors to be explored. This could result in a significant decrease in model run time and give quick results when there is not enough time to complete a full DOE. Continued research and exploration in the use of this factor screening technique could potentially benefit the simulation community.

APPENDIX A. NOLH DESIGN MATRICES

The following three design matrices were considered for use as DOEs for the S4 communications scenario. The designs have been sent to ARL-SLAD and NMSU-PSL to use when they are ready to start the experimentation.

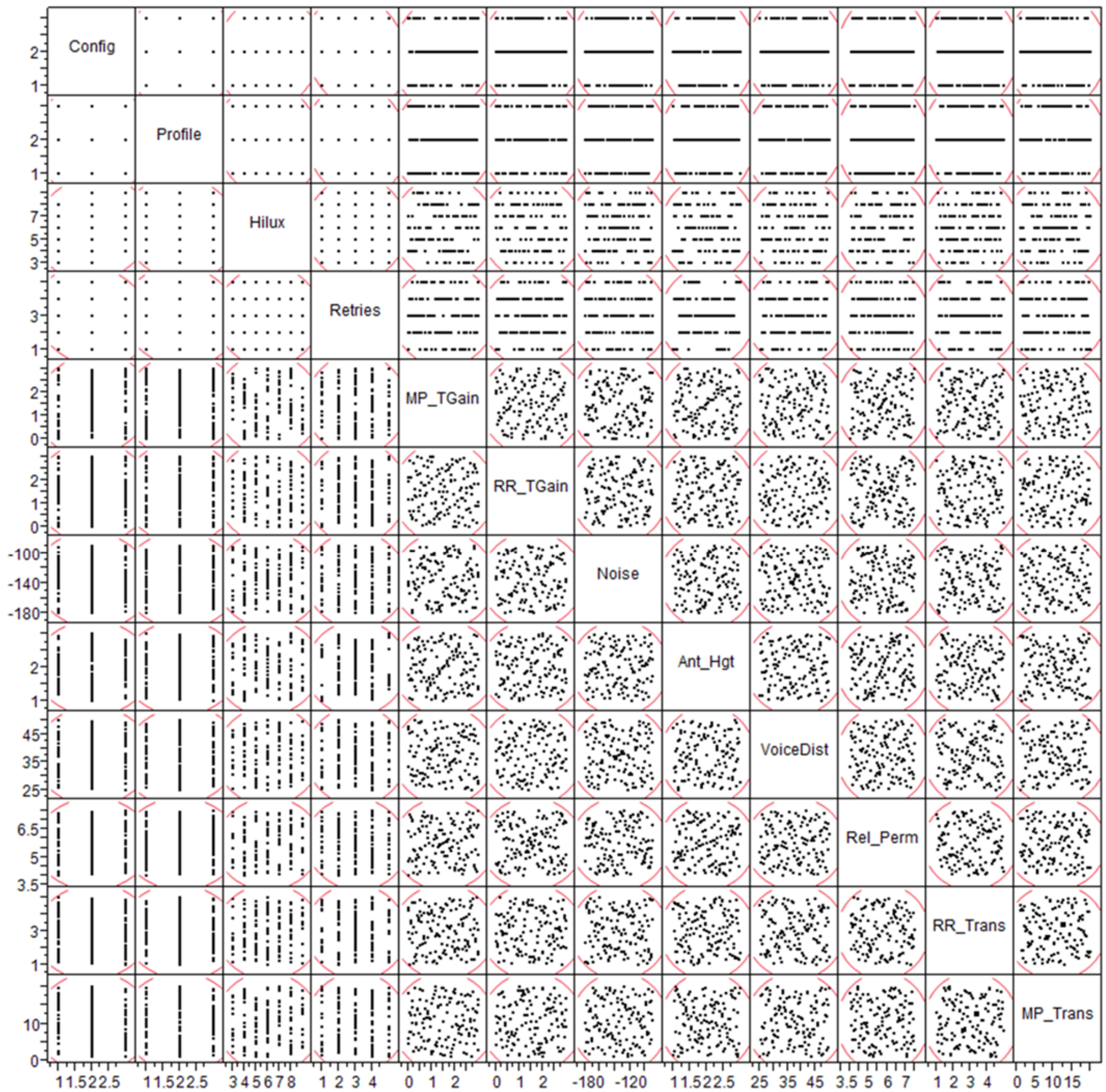


Figure 22. The 129-design point NOLH provides good coverage of the parameter space, but has limited samples in the corners of the parameter space. The maximum pairwise correlation for this design is 13.31%.

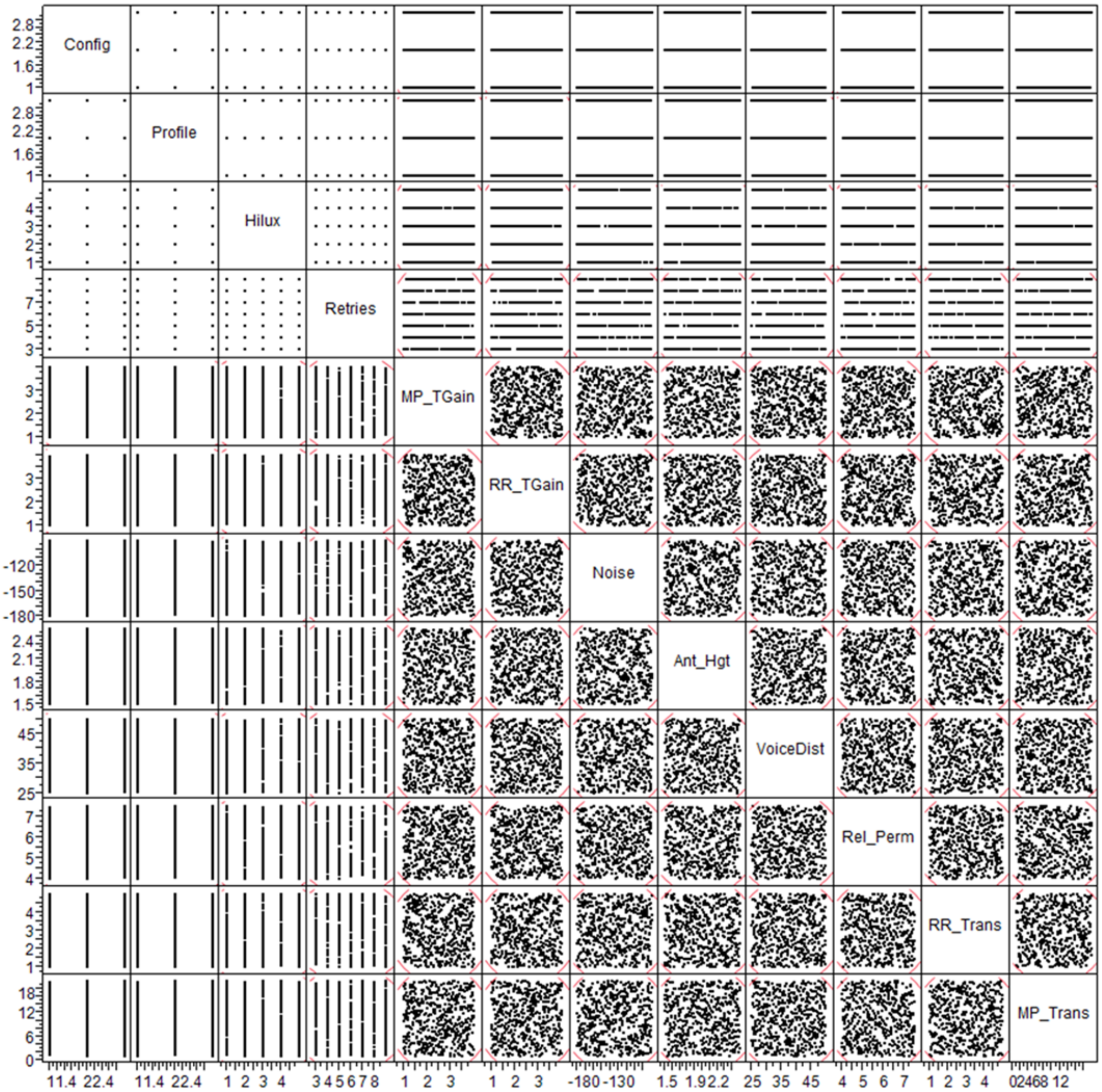


Figure 23. The 512-design point NONBMD provides excellent coverage of the parameter space. The maximum pairwise correlation for the design is 2.37%. This design should have a longer run time than the NOLH, but provides more coverage with a smaller correlation.

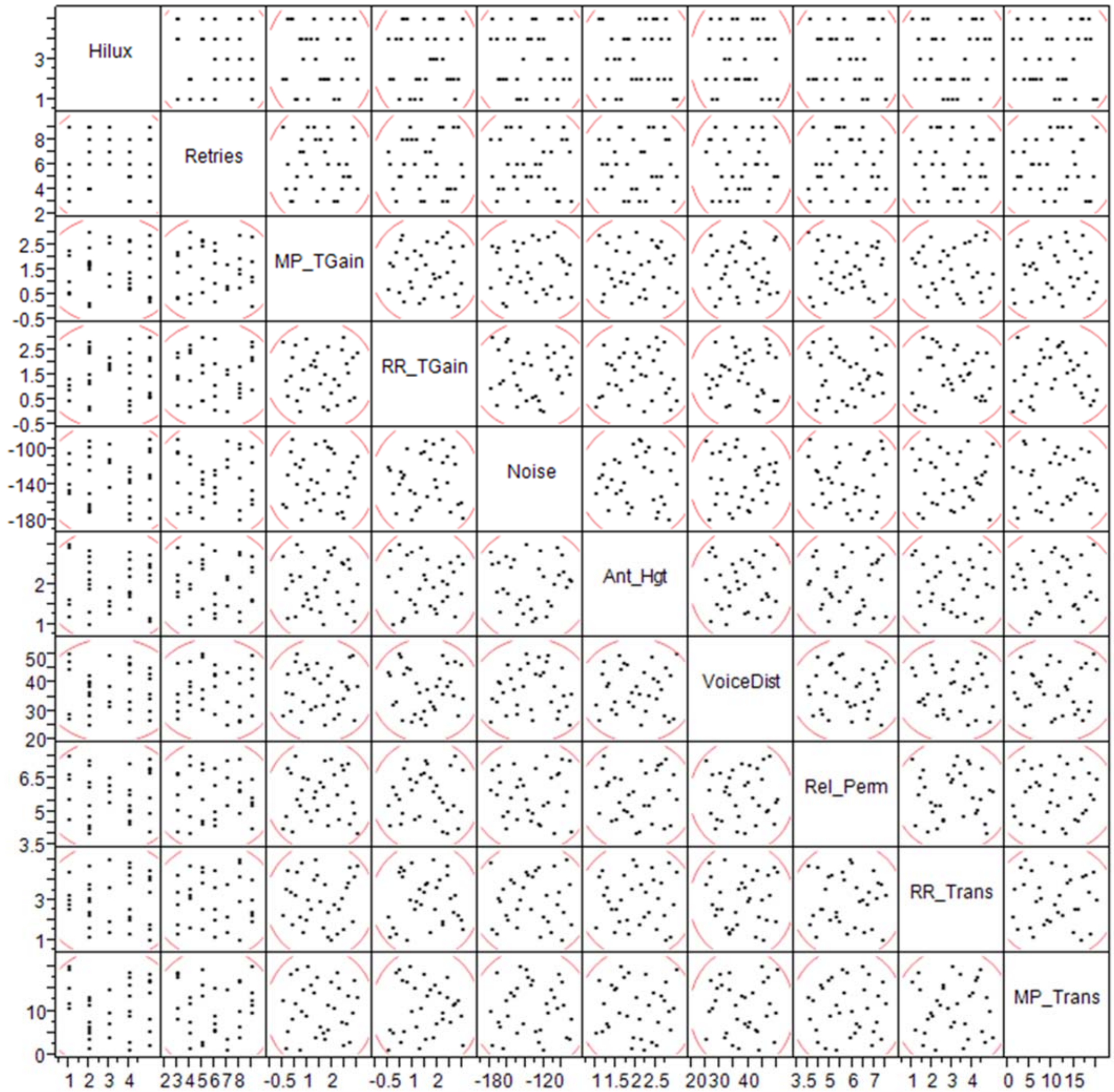


Figure 24. The 33-design point NONBMD created using the DesignCreator (MacCalman, 2012) is less space filling than the other designs, but allows the S4 team to run the model in less time. The design can be run for all combinations of the categorical parameters. The maximum absolute pairwise correlation for this design is 0.9%.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. CODE USED FOR THE FACTOR SCREENING MODEL

```r
myModel <- function(n, mu, s, sdNoise, vars=69, iter=10000, csv=TRUE) {
  library(MASS)
  ## make beta a full-up matrix, with 'iter' rows and 'vars' columns
  beta = matrix(0, iter, vars)
  coef = numeric(vars + 1)
  fname = paste('outputN', n, 'M', mu, 'S', s, 'SD', sdNoise, '.csv', sep='')
  ## initialize the beta *matrix*
  beta[1:iter, 1:n] = rnorm(iter*n, mu, 1)
  results = t(apply(beta, 1, function(vec) {
    response <- designX %*% vec + rnorm(nrow(completeData), sd = sdNoise)
    completeData <- cbind(designX,response)
    completeData[,70] <- response
    colnames(completeData)[70] <- 'y'
    completeData <- data.frame(completeData)
    myform <- as.formula (paste ("~ . +", paste (names (completeData)[-70], collapse="+")))
    modelX <- lm(y~1,completeData)
    modelXstep <- MASS::stepAIC(modelX, myform, direction = "both", steps = s, trace = F)
    coef[c(1, as.integer(gsub('V', '', names(modelXstep$coefficients)[-1])) + 1)] = modelXstep$coefficients
    ## concatenate the input with the output
    c(vec, coef)
  }))
  newData <- matrix(nrow=iter,ncol=n)
  for(i in 1:n) {
    newData[,i] <- results[,i]*results[,(1+vars+i)]
  }
  results <- cbind(results,newData)
  colnames(results) <- c(paste('B', 1:vars, sep=''),
                         paste('C', 0:vars, sep=''),
                         paste('R', 1:n, sep=''))
  ## single write to a file
  if (csv) write.csv(results, fname, row.names=FALSE)
  ## invisible() means that if you capture the function into a variable, you'll get the full results;
  ## otherwise, it won't clobber your display with a humongous matrix
  if (csv) invisible(data.frame(results))
  else data.frame(results)
}
```

Figure 25. R code used for the factor screening model.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Bernstein Jr., R., Flores, R., & Starks, M. (2006). *Objectives and capabilities of the system of systems survivability simulation (S4)* (ARL-TN-260). Las Cruces, NM: Army Research Laboratory.

Buss, A. (2002). Component Based Simulation Modeling with Simkit. In *Proceedings of the 2002 Winter Simulation Conference* (pp. 243–249). San Diego, CA: Institute of Electrical and Electronics Engineers.

Cioppa, T. M., & Lucas, T. W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics, 49* (1), 45–55.

Davidson, J., & Pogel, A. (2010). Tactical agent model requirements for M&S-based IT: C2 assessments. *C2 Journal*, *4* (1), 1–52.

Davidson, J., Pogel, A., & Smith, J. A. (2008). The role of battle command in information system assessments. In *Proceedings from the International Conference on Industrial Engineering Theory, Applications, and Practice* (pp. 154–160). Las Vegas, NV: International Journal of Industrial Engineering.

Devore, J. L. (2011). *Probability and statistics for engineering and the sciences (8th ed.)*. Independence, KY: Cengage Learning.

Gaudard, M., Ramsey, P., & Stephens, M. (2006). *Interactive data mining and design of experiments: the JMP® partition and custom design platforms*. Brookline, NH: North Haven Group.

Hartley, R. (2013, February). *The architecture and workings of the system of systems survivability simulation (S4)* (NMSU-PSL Working Paper). Las Cruces, NM.

Hernandez, A. S., Lucas, T. W., & Carlyle, M. (2012), Constructing nearly orthogonal Latin hypercubes for any nonsaturated run-variable combination, *ACM Transactions on Modeling and Computer Simulation, 22* (4), 20:1–20:17.

Hudak, D., Mullen, J., & Pogel, A. (2008). Determining the impact of information on decision-making in contexts lacking well-defined utility functions. In *Proceedings from the International Conference on Industrial Engineering Theory, Applications, and Practice* (pp. 102–108). Las Vegas, NV: International Journal of Industrial Engineering.

MacCalman, A.D. (2012). DesignCreator spreadsheet. Available online via http://harvest.nps.edu (accessed 02/06/2013).

Miller, N.L., & Shattuck, L.G. (2004). A process model of situated cognition in military command and control. In *Proceedings of the 2004 Command and Control Research and Technology Symposium*. San Diego, CA: Space and Naval Warfare Systems Center (SPAWAR) and San Diego Marine Recruit Depot.

Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics(OUSD[AT&L]). (2009, December 9). *DoD modeling and simulation (M&S) verification, validation, and accreditation (VV&A)* (DoD Instruction 5000.61). Washington, DC: Author.

Sanchez, S. M. (2011). NOLH designs spreadsheet. Available online via http://harvest.nps.edu [accessed 02/06/2013].

Sanchez, S. M., & Wan, H. (2009). Better than a petaflop: The power of efficient experimental design. In *Proceedings of the 2009 Winter Simulation Conference* (pp. 60–74). Austin, TX: Institute of Electrical and Electronics Engineers.

Smith, J. A., Bernstein Jr., R., Hartley, R., & Harikumar, J. (2012). System of systems analysis (SoSA) introduction. ARL-SLAD and NMSU-PSL Presentation.

Starks, M., & Flores, R. (2004). *New foundations for survivability lethality vulnerability analysis (SLVA)* (ARL-TN-216). Las Cruces, NM: Army Research Laboratory.

Vieira Jr., H. (2012). NOB_Mixed_512DP_template_v1.xls design spreadsheet. Available online via http://harvest.nps.edu [accessed 02/06/2013].

Vieira Jr., H., Sanchez, S. M., Kienitz, K. H., & Belderrain, M. C. N. (2012). *Conducting trade-off analyses via simulation: Efficient nearly orthogonal nearly balanced mixed designs*. Working paper, Operations Research Department, Naval Postgraduate School, Monterey, CA.

Xing, D., Wan, H., Zhu, M.Y., Sanchez, S.M., & Kaymal, T. (2013). Simulation screening experiments using Lasso-Optimal Supersaturated Design and Analysis: A maritime operations application. In *Proceedings of the 2013 Winter Simulation Conference* (pp. 497–508). Washington, DC: Institute of Electrical and Electronics Engineers.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California